Topologic Navigation and the Pfafstetter System

By

Jordan Ernest Furnans, BSE

Thesis

Presented to the Faculty of the Graduate School

of The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December, 2001

Topologic Navigation and the Pfafstetter System

APPROVED BY

SUPERVISING COMMITTEE:

__________________________, Supervisor
David R. Maidment

____
Ben R. Hodges

Á minha linda esposa Debora, quem me fornece sempre com mais amor e carinho que eu mereço.
Não posso imaginar um dia sem você.
Sempre estarei ao seu lado pelo amor!

ACKNOWLEDGEMENTS

I was once told by a retiring professor that you never know quite what life will bring you. My past few years have definitely supported that statement, both in terms of my personal and academic lives. Here I want to give a brief thanks to the many people who helped me through everything…..

To my advisors in academia, Dr. David Maidment, Dr. Francisco Olivera, and Dr. Ben Hodges: Thanks for providing me the opportunity to explore my own research interests, and for providing me with guidance that only years in the field can bring. I look forward to continuing academic relations with all of you, for there are many interesting topics out there to be investigated. Oh, and I look forward to traveling about the world with you all, presenting these topics at all corners of the globe. The net stop is Australia.. After that, how about Peru? Brazil? New Zealand?

To my fellow EWRE grad students, with whom I could always share triumphs and frustrations. We always learn best together. Thanks Adriano, Dan Tim, Vicki, Sarah, Kevin, Melissa, Venkatesh, Kim, and all the others who played a part.

To my family, both nuclear and extended. You were always inquisitive and understanding, always ready to give support. Mom, Dad, Jo – thanks the most. I only wish Texas were closer to you so that we could share more of these wonderful experiences. Uncle Rick – maybe I should have gone to Stanford… Nah, I think I'm doing more than fine here.

Finally, to the biggest surprise in my life, to the woman who makes me smile wider than I ever though possible. Deb, now that this silly thing is done, lets

!!!!!!CELEBRATE!!!!!!

Topologic Navigation and the Pfafstetter System

By

Jordan Ernest Furnans, MSE.

The University of Texas at Austin, 2001
SUPERVISOR: David R. Maidment

ABSTRACT

This thesis explores the spatial structure of drainage area connectivity. In particular, it describes a new method for tracing water movement from one drainage area to the next through the landscape. This method, named Topologic Navigation, is applicable on drainage area datasets whose entries each have as attributes a unique ID code and the ID code of the area immediately downstream. From this information, the areas immediately upstream of each area in the dataset are discernible. With knowledge of this connectivity, it is possible to derive the entire upstream and downstream drainage areas of a single area in the landscape.

Downstream area IDs may be determined automatically for datasets attributed with Pfafstetter-based codes. These codes are assigned to drainage areas based on the relative drainage topology of the areas. A detailed methodology for determining downstream IDs using Pfafstetter-based numbering systems is developed. Use of the downstream IDs in performing Topologic Navigations is also demonstrated and discussed. Extensive modifications to the existing Pfafstetter-based numbering systems are also suggested.

Table of Contents

Table of Contents, CONT.

LIST OF FIGURES

LIST OF FIGURES, CONT.

LIST OF FIGURES, CONT.

LIST of TABLES

Chapter 1 – Introduction

One of the most characteristic features of the landscape is the drainage topography. This pattern of ridges, slopes, drainage paths and streams is visible upon looking at the landscape, especially when viewed from the air. Hydrologists have designed methods to analyze drainage topography and to delineate layouts of watersheds or drainage basin boundaries. This has been done for all of the United States within the Hydrologic Unit dataset (Seaber et al., 1987; USGS-1, 2001) and for the entire world within the HYDRO1K dataset (EROS-1, 2001; Verdin and Verdin, 1999). A characteristic of these watershed layouts is that they each form a network, with one drainage area flowing into the next area downstream, and so on until a terminal basin (e.g. an enclosed lake) or the ocean is reached. This thesis explores the spatial structure of drainage area connectivity. In particular, it explores methods for tracing water movement from one drainage area to the next through the landscape, and for using this connectivity to derive the entire upstream and downstream drainage areas of a single area in the landscape.

Watershed managers often need to understand how engineering applications in one watershed might affect the water usage in another watershed. For example, before constructing a hydro-electric dam, the upstream catchment area of the dam must be known. This upstream area will be partially inundated as a result of the dam, and the affects of and extent of this inundation must be known. The areas downstream of the dam are important, because the river flow through this area will be altered due to the dam's presence. If the dam were to rupture, the resultant flood wave would likely cause damage in downstream watersheds. Identification of the potentially threatened land areas is crucial in developing disaster mitigation strategies.

A second example is related to water quality and pollutant transport. As flows are necessarily downstream, contaminant detection is always downstream of the source. Thus, the possible source locations are limited to the upstream drainage basin, while subsequent transport is confined to the downstream basin. Computing the downstream and upstream regions relative to any particular point, therefore, has direct utility to water system managers.

The process of tracing water movement through the landscape may be broadly referred to as "navigation" in the sense that it is possible to navigate upstream and downstream along the flow path of a water particle. Many computerized and manual navigation techniques have been developed that serve to identify such flow paths. The computerized methods often involve the use of Geographic Information Systems (GIS) technology to determine and describe the relative topology of sections of the landscape. Each navigation methodology, however, differs with respect to how topologic information is derived from the input data.

The network analysis tool in the ArcGIS software makes use of a river network to determine landscape topology. With this tool, it is possible to navigate upstream and downstream along a vector stream network, with the navigation originating from any junction on the network. The river network implies the landscape topology, because rivers always carry water in the down-slope direction. However, because the navigation is performed on a river network, in order to identify the actual watersheds upstream of a given junction, one must select the watersheds that encompass the stream segments identified in the network navigation. A second method, also requiring the ArcGIS system but specifically the Spatial Analyst extension, uses raster (grid) data to perform navigations. Elevation data for a landscape may be stored in special grids, and through special grid processing algorithms the landscape topology may be determined. With this extension and the elevation grid data, the user selects a point on the landscape, and the area upstream and downstream of that point is identified.

The Topologic Navigation technique, as introduced in this work, is a new navigation technique that is directly applicable to vector watershed datasets. This technique derives topologic relationships between drainage areas from attributes of the drainage areas. Specifically, each drainage area must store as attributes a unique identification code and the identification code of the area downstream of itself. The data user is required to determine the downstream identification codes, and the methodology for doing so is up to the user. However, many watershed datasets produced by the U.S. Geological Survey (USGS) already contain or will contain downstream attributes for each entry. One example is the National Watershed Boundary dataset (USDA, 2001), which is a more detailed version of the HUC dataset for the United States. Another example is any dataset that will be formatted according to the ArcGIS Hydro Data Model framework (Maidment, 2001). This framework is supported by the USGS and the US Environmental Protection Agency (EPA), and it may eventually become the standard framework for all hydrologic datasets. Therefore, the Topologic Navigation technique may soon be directly applicable to the majority existing datasets.

The Topologic Navigation technique is also unique in that the results from the navigation process are incorporated as new attributes in the dataset. Other existing navigation methods merely distinguish the upstream and downstream elements from other elements in the dataset, therefore forcing the user to create new files for each set of navigation results. Also, by including the navigation results in the dataset attribute table, these results become accessible to other programs than may simulate hydrologic processes. In this way, the topologic navigation process can be used as a support process for water quality and water availability models.

For existing datasets that do not contain downstream area codes, the Topologic Navigation technique is not directly applicable. However, if the datasets are attributed based on the principles of the Pfafstetter system (Pfafstetter, 1989; Verdin and Verdin, 1999; Silva, 1999), then the downstream area codes may be automatically determined. In this classification system, watersheds and rivers are assigned unique ID numbers based on their topologic relations with surrounding features. Determination of the downstream area does not require any spatial analysis capabilities, because the landscape topology is implied by the Pfafstetter codes. A detailed study of the Pfafstetter system was conducted as part of this work, and a methodology was developed for determining downstream areas based on an analysis of Pfafstetter-codes. This methodology and that inherent in the Topologic Navigation technique were implemented into a suite of Microsoft Excel macros, referred to as the “Pfafstetter Tools.” These macros were applied to the attribute tables of HYDRO1K datasets, which are the only currently available datasets attributed with Pfafstetter-based codes.

The downstream area determinations and navigation results on the HYDRO1K datasets demonstrated the applicability of the Pfafstetter and Topographic Navigation methodologies. However, the results also identified inconsistencies between the Pfafstetter system principles and the drainage area codes within the HYDRO1K dataset. These inconsistencies may be ameliorated through the implementation of various modifications to the Pfafstetter system. These modifications, if implemented, will enhance the topologic references implicit within datasets attributed with Pfafstetter-based codes. Such datasets would further enhance the applicability of the Topologic Navigation technique.

This work provides a detailed description of the methodology behind the Topologic Navigation technique, as well as an in-depth analysis of the Pfafstetter-based methodologies for numbering drainage areas. Chapter 2 details the existing tracing methods within the ArcGIS system, the existing methods for assigning codes to drainage areas, as well as an existing navigation method that makes use of the Pfafstetter system. Chapter 3 explains the methodology of the Topologic

Navigation technique, and Chapter 4 describes the topologic characteristics of the Pfafstetter system. Chapter 5 is an in-depth discussion of the implementation of the Topologic Navigation methodology and Pfafstetter-based methodology for downstream area determination into macros for use on the HYDRO1K data. The results of multiple applications of the Topologic Navigation technique is discussed in Chapter 6. Also discussed are possible modifications to the Pfafstetter system that would enhance the utility of the system and the Topologic Navigation technique. Conclusions drawn from this work are given in Chapter 7, as is a discussion of work yet to be completed in this field. The Visual Basic™ program code for each module is provided in Appendix A, and all interested parties are invited to use and modify the code for their own purposes. As discussed throughout this work, many improvements to the Pfafstetter System and the Topologic Navigation technique are possible.

Chapter 2 – Literature Review

Topological connectivity between drainage areas has always been an interest for cartographers and watershed managers (Newson, 1992). In the years past, interested parties would use topographic maps to manually determine drainage area boundaries and the interrelations between drainage areas (Barrow, 1998). Recently, with the invention and widespread use of electronic maps and Geographic Information Systems (GIS) software, many new automated techniques for producing the same type of results have been established (Goulter and Forrest, 1987; Burton, 1995). These automated techniques are well documented, and were initially based on raster topographic data (Jensen and Domingue, 1988). Newer programs and methodologies have been developed to determine topologic relationships between hydrographic features represented as vector data (Verdin, 2001; Furnans and Olivera, 2001). Topologic Navigation, which is introduced in this thesis, is one of these vector-based techniques, although it shares its roots in both the raster based and vector based navigation methods.

2.1 Describing Topology with Raster Data

The standard method for delineating drainage areas with GIS software involves the use of a digital elevation model, or DEM. DEMs are digital records of terrain elevations for ground positions at regularly spaced horizontal intervals. These grids are derived from standard topographic quadrangle maps through the use of hypsographic data and /or photogrammetric methods (USGS-2, 2001). Such grids are easily processed with the ArcGIS hydrology tools in the Spatial Analyst extension. The following discussion describes the basic theory behind the watershed delineation functions in ArcGIS.

The grid operations involved in watershed delineation are all derived from the basic premise that water flows downhill, and in so doing it will follow the path with the largest gradient (steepest slope). In a DEM grid structure there exist, at most, eight cells adjacent to each individual grid cell. Accordingly, water in a single cell may travel in one out of at most eight different directions in order to enter the next downstream cell. This concept is referred to as the 8-direction pour point model (ESRI, 2000; Furnans and Olivera, 2001).

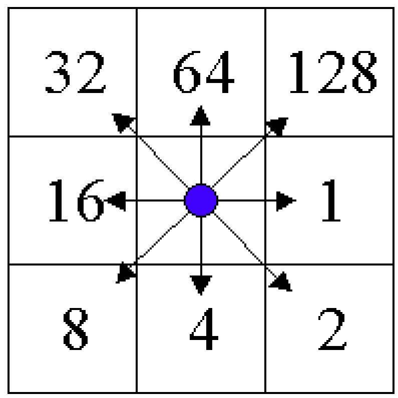


Figure 2.1 – The 8-Direction Pour Point Model for Grid Operations

In this grid representation, water in a grid cell may flow only along one of the eight paths depicted by arrows (Figure 2.1). The number in each cell represents the direction water travels to enter the nearest downstream cell, and the numbering scheme has been set by convention (ESRI, 2000). The numbers were determined from the series $2^x \quad x = \{0,1,...7\}$, which reflects the binary form of numbers in computer systems.

Watershed delineation with the 8-direction pour point model is best explained through illustration. For demonstration purposes, a section of a sample DEM grid is given. The numbers in each grid cell represent the cell elevation.

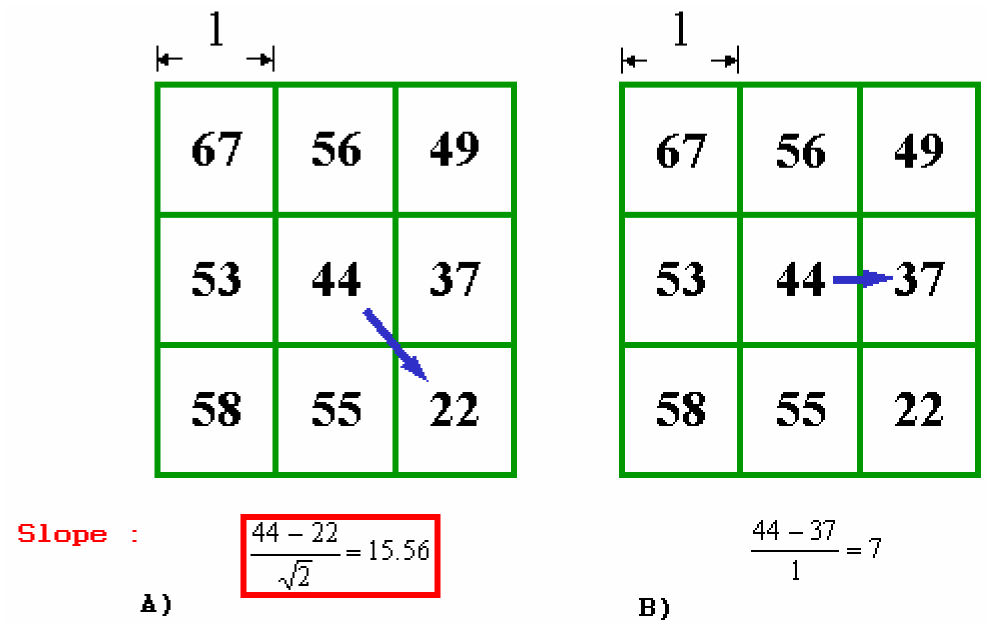


Figure 2.2 – Slope Calculations with the 8-direction Pour Point Model – A) Slope calculated for diagonal cells; B) Slope calculated for cells with common sides.

Focusing on the center cell (value = 44) as shown in Figure 2.2, only 2 of the 8 adjacent cells contain elevation values less than 44, thereby limiting the possible flow directions in that water will not flow to a cell with a greater elevation. A fundamental assumption is that water will only flow in the direction in which the greatest elevation decrease per unit distance is obtained. In Figure 2.2a, this slope is calculated along the diagonal by subtracting the destination cell value from the original cell value, and dividing by $\sqrt{2}$, the distance between the cell centers assuming each cell is 1 unit long on each side. In Figure 2.2b, the slope is calculated to the non-diagonal cell. It is equal to the elevation difference because the distance between the cell centers is unity. In this case, the diagonal slope is greater, and water will flow toward the bottom right cell. The center cell is then assigned a flow direction value of 2. This process is then repeated for each of the cells in the DEM grid, and a new grid is created to store the results of the calculations. This new grid, called the Flow Direction grid (ESRI, 2000), contains cells with only the numerical values dictated by the 8-direction pour point model.

67	56	49	46	50
53	44	37	38	48
58	55	22	31	24
61	47	21	16	19
53	34	12	11	12

A)

2	2	4	4	8
1	2	4	8	4
128	1	2	4	8
2	1	4	4	4
1	1	1	2	16

B)

Figure 2.3 - Grid Operations – A) DEM Grid; B) Flow Direction Grid. Note: Area in red is from the previous figure.

The flow direction grid (Figure 2.3) represents the flow topology of the landscape. This topology is evident when the numbers in the grid are represented by arrows whose direction is dictated by the 8-direction pour point model (Figure 2.4a). These arrows may also be represented as a flow network (Figure 2.4b). The upstream and downstream connectivity can be readily determined from this flow network. Therefore, the topological relationships between grid cells are evident.

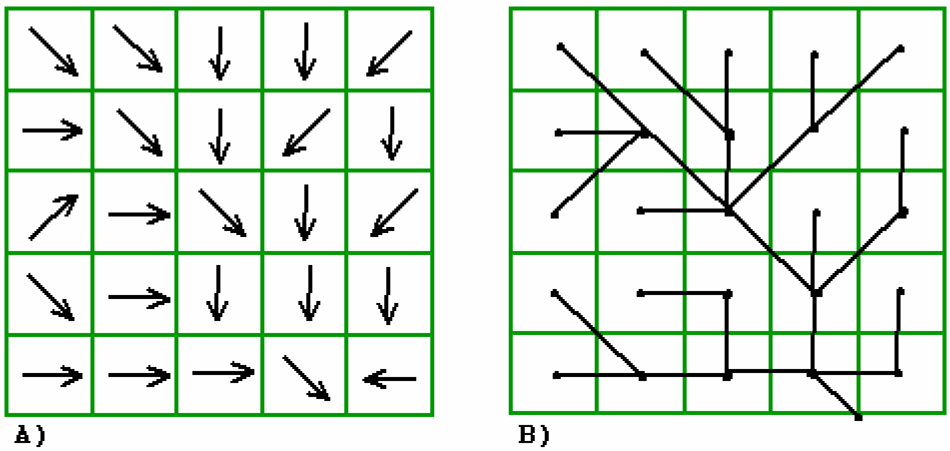


Figure 2.4 - Physical Representations of a Flow Direction Grid –A) with directional arrows; B) As a flow network

A flow accumulation grid is calculated from the flow direction grid. This grid records the number of cells that drain to each single cell in the grid, and is therefore a measure of how far “downstream” a cell is from water sources. It is also a measure of the relative magnitude of flow that may accumulate in each cell (Figure 2.5).

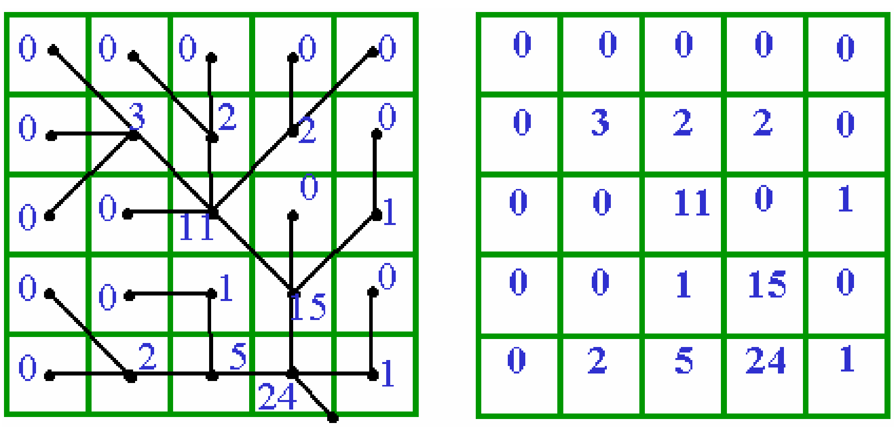


Figure 2.5 - Flow Accumulation Grids – grids storing the number of cells draining to a given cell (blue) along the flow network

Streams may be defined through the use of a threshold value applied to the flow accumulation grid. For example, if a value of 5 were set as the threshold, than any cell with a flow accumulation greater than 5 would be considered a stream. Cells with flow accumulations greater than or equal to the threshold are given a value of 1 in a newly created Stream Grid, with all other cells containing the value 0 (Figure 2.6).

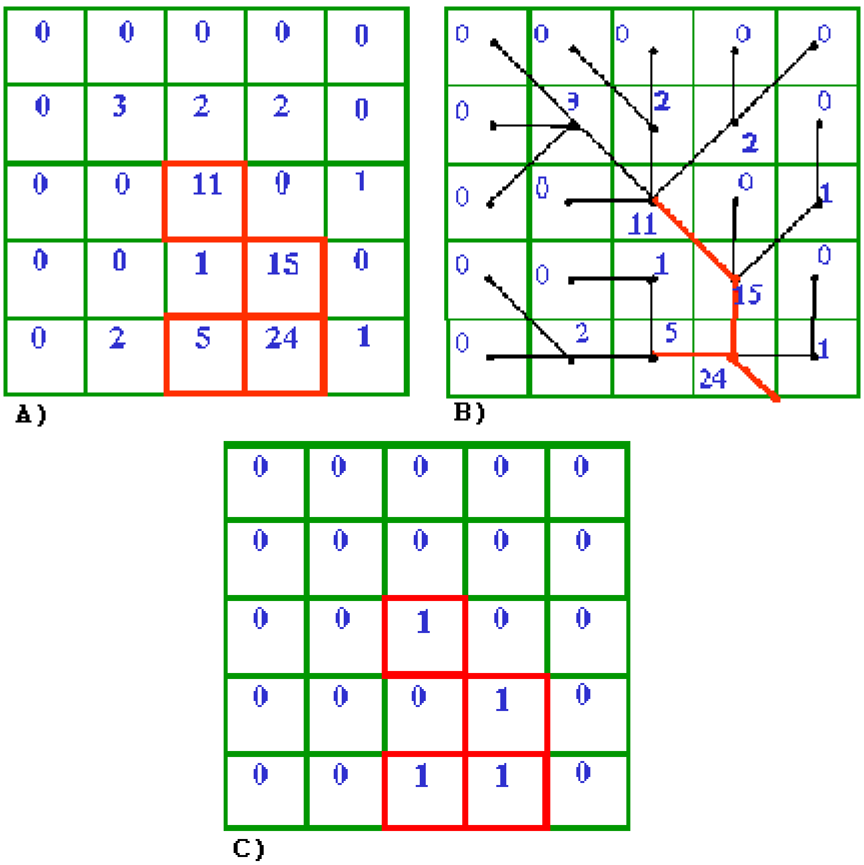


Figure 2.6 - Stream Definition from the Flow Accumulation Grid and a Threshold Value – A) Grid cells with accumulation greater than or equal to 5 are considered stream cells (red); B) Streams identified on the flow network (red); C) Stream Grid

The created stream networks may be divided into distinct stream segments, which is useful in delineating multiple drainage areas. If only the overall watershed is desired, the delineation function could be used on the established grids as long as the outlet cell is defined. For this discussion, multiple drainage areas are delineated.

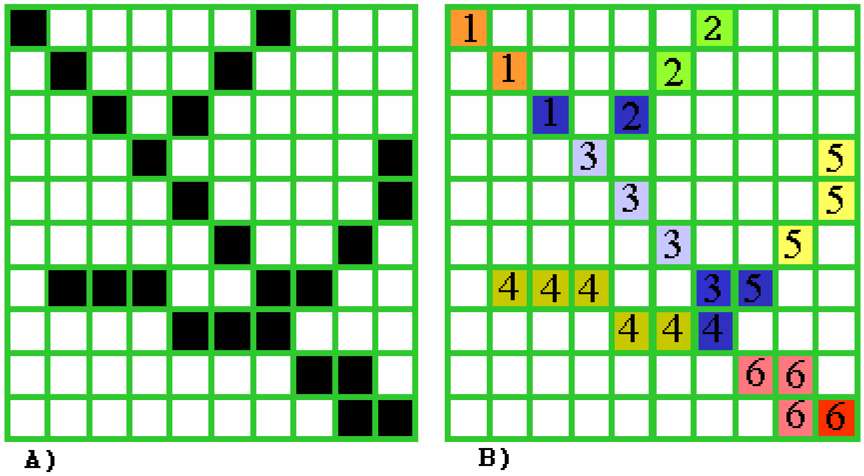


Figure 2.7 - Stream Links Defined – A) Stream Grid representation, B) Stream Links (numbers) defined, link outlets (blue), watershed outlet (red)

Each cell within a stream link is assigned the same number (Figure 2.7). The most downstream cell in each link is the link outlet cell, which is represented as a drainage point in the data model. An outlet grid, with the individual outlets cells (in Figure 2-7b**,** the blue or red cells) containing the stream link codes and all other cells containing NODATA, is then produced from the streamlink grid. Stream link codes can be used as drainage ID values when the resulting drainage areas are determined.

The drainage areas are delineated through the use of the flow direction and outlet grids. The procedure involves the determination of all of the cells that drain to each outlet. It assigns each of these cells the value of the outlet cell (which also may correspond to the values for the cells in the stream link grid). The delineation results are stored in a watershed grid (Figure 2.8).

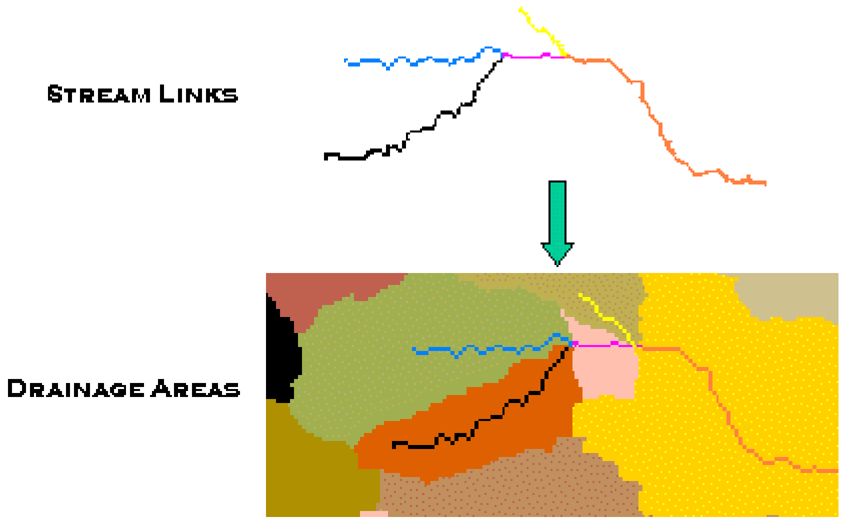


Figure 2.8 – Delineation Results – One drainage area for each stream link

The watershed grid only describes areas that drain to specific streamlinks in the streamlink grid. Each area does not "know" which areas are immediately downstream and immediately upstream of itself. However, this information is contained within the flow direction grid, and by inspection of this grid, the topological relationships between drainage area may be determined.

This intersection of the watershed grid and the flow direction grid in order to determine topology is currently included in the Hydrologic Modeling functionality of the ArcGIS Spatial Analyst extension (ESRI, 2001). The updated version of this extension, which works with the new ArcGIS software, allows the user to determine the topology about a selected point on a watershed grid. The function then uses the flow direction grid to select/highlight all of the grid cells upstream and downstream of the selected cell (ESRI, 2001; ESRI-1, 2001).

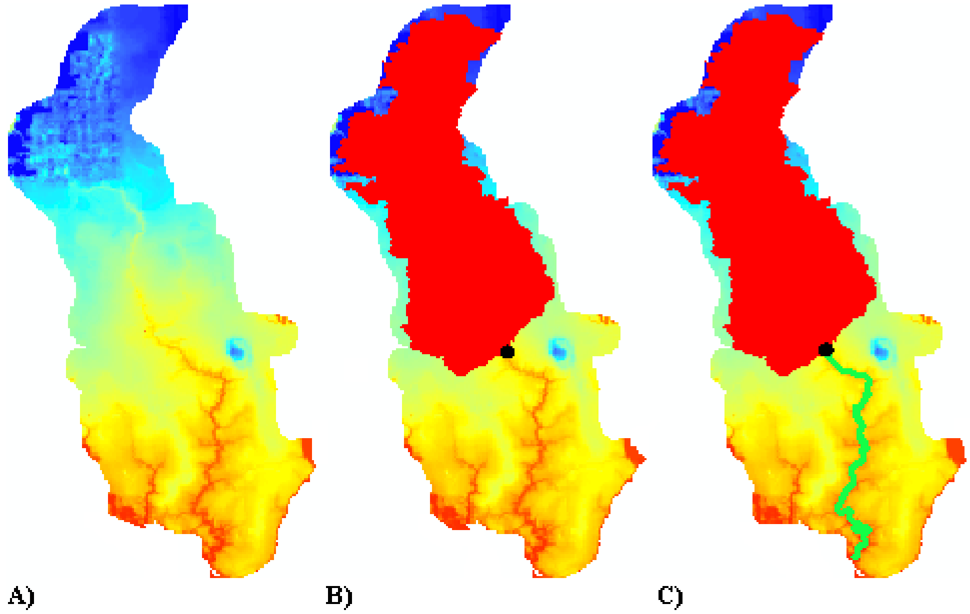


Figure 2.9 – Raster-based Navigation with the Hydrologic Modeling Extension – A) DEM grid, B) Upstream watershed (red) of a selected cell (black), C) Downstream flow path (green) from the selected cell (black).

The Hydrologic Modeling functionality of the ArcGIS Spatial Analyst extension produces visual results of the topology around a selected point. However, the extension is only viable within the ArcGIS environment, with the Spatial Analyst extension included. Also, it requires raster data, which is often overwhelming in size and difficult to obtain, depending on the area of interest.

2.2 Determining Watershed Connectivity with Vector Data

Most often, drainage areas derived by raster processes are converted into vector polygons for further processing. Vector data is more manageable than raster data in that it is often less memory intensive. Vector data, which consists of polygons, lines, and points, may contain multiple attributes. These attributes may be a mixture of numerical and alphabetical values as necessitated by the data user. In order to contain more attributes and enhance data processing, the raster-based streams and drainage areas may be vectorized based on the ID field. Therefore, each distinct stream link in the stream link grid becomes a distinct vector arc, and each drainage area in the watershed grid becomes a distinct vector polygon. The ID value from the respective grid is transferred to the ID value of the corresponding vector object.

A drawback of the vector format is that the topologic information from the flow direction grid is lost in the vector data. The vectorization process does not transfer any direct topological information from the raster to vector formats. The resulting polygon drainage areas do not "know" to which other polygons they contribute flow.

Within the vector format, the topologic relationships may be determined based on the streams, which are stored as linear arc features. Streams always flow down a gradient, from a higher elevation to a lower elevation. Therefore, by determining the topologic relations among the stream segments, the relationships between the areas draining to these segments are obtainable. This process is also possible with the ArcGIS software, but unlike with the raster data, determining vector-based topologic relationships requires the Network Analysis extension (ESRI-1, 2001).

The Network Analysis extension is based on the concept of a geometric network, which is a series on interconnected linear features (ESRI-1, 2001). Such networks could include representations of city streets, or they could include representations of a set of streams. For hydrologic modeling purposes, the streams defined from the raster data could be built into a geometric network within the ArcGIS system. Topology is evident on such a network as long as the network outlet is identified. This outlet is referred to as a "sink," and it is the most downstream location along the network. All streams and points on the network must eventually flow to the sink, and therefore the topology of the system may be determined.

One drawback of the Network Analysis extension is that it can only determine the topology of linear features. The topology of areas about these features must be determined by association. Specifically, the Network Analysis extension only selects those arcs upstream and/or downstream of a given point. The drainage areas related to these points must be selected based on the “Select by Location” query function within the regular ArcGIS system.

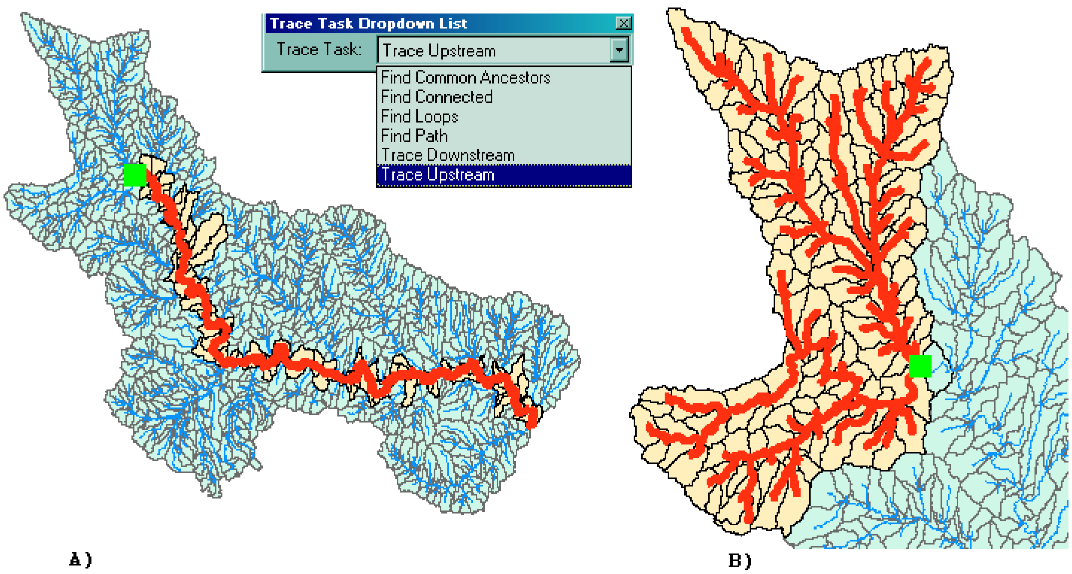


Figure 2.10 – Vector Topology with the Network Analysis Extension A) downstream trace, B) upstream trace. Network Analysis toolbar is also shown in part.

In Figure 2.10, each trace is carried out on a geometric network from the point identified by the square. The trace only identifies the appropriate network arcs (shown in bold red) relative to the selected point. The drainage areas containing these arcs (tan polygons) must be selected from the drainage area layer by their geographic proximity to the trace-identified arcs. After the selections are made, the drainage areas may be converted into new data layers (ArcGIS shapefiles or coverages) and then manipulated by the user. With this method, it is possible to create new vector layers that describe the topology about a certain point. However, if it becomes desirable to determine the topology about a second point, the tracing must be performed again. Also, it is impossible to automatically determine the area immediately downstream of all areas in the database by using the Network Analysis extension. Therefore the utility of this extension depends on the results desired by the data user. Also, the user must have both a stream network and a set of drainage areas in order to determine drainage area topologies with the Network Analysis extension. This is unlikely to be a problem if the drainage areas are raster-based, but it is a potential difficulty if the areas were determined by other means.

2.3 Drainage Area Numbering Methodologies

The methods for determining relative drainage area topologies for both raster and vector data each require the recently developed ArcGIS software package. The previous GIS software packages from which the ArcGIS system was derived do not support the hydrologic modeling component of the Spatial Analyst extension or the Network Analysis extension. Therefore, users with older, more familiar versions of GIS software do not have the capability to determine drainage area topology. Few attempts have been made to develop a methodology for determining topology within the older ArcView and ArcInfo GIS systems. None of these attempts resulted in published work. The most promising of these attempts involves methods for assigning ID values to hydrographic data elements.

Every drainage area in a dataset usually contains some type of unique attribute that identifies the area from all other database entries. This attribute is often a unique numerical ID, which may be assigned according to any number of methods. Various methodologies exist for numbering drainage areas in logical and loosely topologically-based manners. For example, stream gauges are each assigned numbers, and the areas upstream of these gauges are often referred to by the number of the stream gauge (Verdin and Verdin, 1999). Many governments and government agencies have developed intricate numbering systems for the drainage areas under their jurisdiction.

One numbering methodology is that of the French research organization ORSTOM for locating their stream gauges. This methodology assigns nine-digit numbers to drainage areas in Africa, South America, Europe, and Oceania (Roche, 1968; Verdin and Verdin, 1999). Each of these nine digits represent different locations across the globe. The first digit identifies the continent, and the second and third digits identify the country in which the area is located. The next two digits represent the river basins on each continent. The final two digits uniquely identify the gauge station location along the river identified by the sixth and seventh digits. All of the correspondence between digits and geographic locations are determined by ORSTROM, and this numbering system has only been defined for the areas in which they manage gauges. Also, the river location digits are organized alphabetically and without regard to the topology of the landscape. Therefore navigation based on the ORSTROM numbers is not possible.

A more topologically related numbering methodology is that of the Hydrologic Unit System. The Water Resources Division of the U.S. Geological Survey (USGS) developed the system for organizing drainage areas for the continental United States (USGS-1, 2001; Seaber et al., 1987). This method describes drainage areas in a hierarchal scheme. Areas are separated into 4 classes: Region, Sub-Region, Basins, and Sub-basins. Each class describes a successively smaller land area. Each Region (the largest area class) is referred to with a two digit Hydrologic Unit Code (HUC). The Sub-Regions contained within each Region are referred to by 4-digit HUCs, with the first two digits equal to that of the Region. In this scheme, Basins have 6-digit HUCs, and Sub-basins have 8-digit HUCs. Figure 2.11 shows the Regions of the United States.

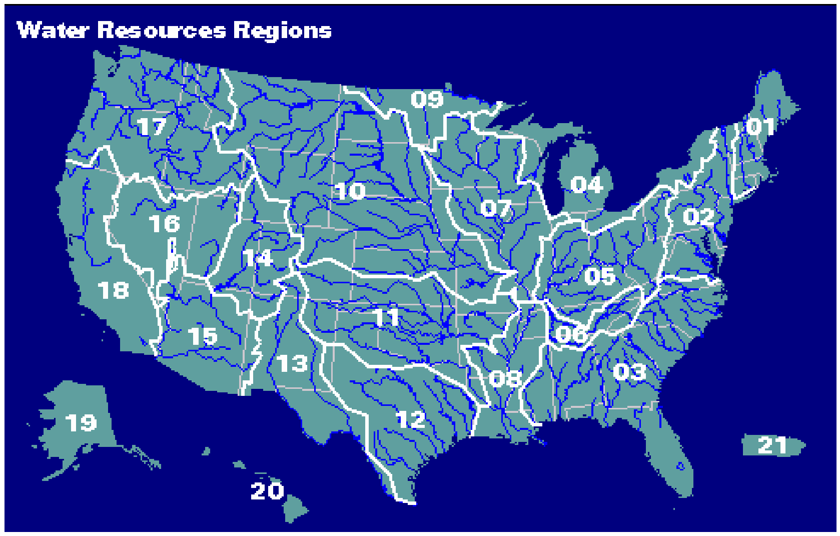


Figure 2.11 - Hydrologic Units – Regions of the United States (USGS-1, 2001)

Under this scheme, the numerical ID of an area is similar to the ID of a neighboring area, although the ID values do not reflect any topological relationships between the areas. Instead, the ID values reflect geographical relationships, which may or may not coincide with the topological relationships between areas on the land surface.

The US National Water Information System (NWIS) maintains a numbering system with an even greater topologic foundation than the Hydrologic Unit system. This system is used to reference the stream gauges from which the NWIS stream flow records are created (Wahl et al, 1995; USGS-3, 2001). Under this "NWIS" system, the eight-digit station numbers decrease in the downstream direction. However, the numbers are not sequential, and gauge #33333333 is not immediately downstream of area #33333334. Ostensibly, gaps are included in the downstream-numbering sequence so that future stream gauges could be added along the river without requiring a re-numbering of all stations along the river.

Each of these numbering systems makes greater use of the geographic locations of their "targets" (Drainage areas, stream gauges, etc.) than the topologic relationships between target locations. Therefore, these systems are inadequate for navigational purposes. However, the Pfafstetter system for drainage area codification focuses entirely on the topologic relationships between areas in order to assign ID codes.

2.4 Topology Implied by the Pfafstetter System

The Pfafstetter system was developed in 1989 by Otto Pfafstetter, a engineer working with the Departamento Nacional de Obras de Saneamento (DNOS), a now defunct government agency in Brazil. The system was designed to organize the drainage areas of Brazil in order to more efficiently distribute irrigation waters (Silva, 1999). This system assigns numerical ID values to streams and drainage areas that are organized in a hierarchal structure. Through comparing the assigned Pfafstetter ID values, the topological relationships between corresponding drainage areas are easily determined (Verdin, 1997; Verdin and Verdin, 1999; Pfafstetter, 1989; Silva, 1999).

In order to assign topologically-based numerical ID's, the total area upstream of each area in the dataset, as well as knowledge of the direction of flow in each river, are required. Both of these sets of information are available from flow direction and flow accumulation grids created within the raster-based drainage area delineation process. An in- depth explanation of how this information is used in assigning Pfafstetter codes is given in Chapter 4.

Researchers at the USGS EROS Data Center have successfully developed a methodology for describing vector data topology based on Pfafstetter codes. This methodology is incorporated into a series of Avenue language scripts that run in conjunction with an ArcView 3.x project. The following discussion is derived from personal communications Kris Verdin of the EROS Data Center and personal tests of the methodology within ArcView 3.2 (Verdin, 2001). The methodology is referred to as the "EROS program."

With the EROS program, the user may determine either the areas upstream of a given area, or the stream segments downstream of a given area. The areas downstream of a given area must be determined by their association with the downstream river segments. The program identifies upstream areas and downstream river segments by selecting those areas through a specialized database query. As with trace queries from the Network Analysis function, the results must be converted to another shapefile in order to be of use to the data user.

As shown in Figure 2-12, the EROS program selects the areas upstream of the user-selected area. This selection is not specific to the hierarchal levels inherent in the Pfafstetter system, and the results are always displayed for the highest hierarchal level supported by the dataset (Higher levels indicate smaller drainage areas – See Section 4.1). Therefore, if the highest Pfafstetter level is level 6, and the user is interested in determining the areas upstream of a level 4 area, the level 6 drainage areas selected by the program must be intersected with the level 4 drainages areas in order to determine the areas of interest to the user.

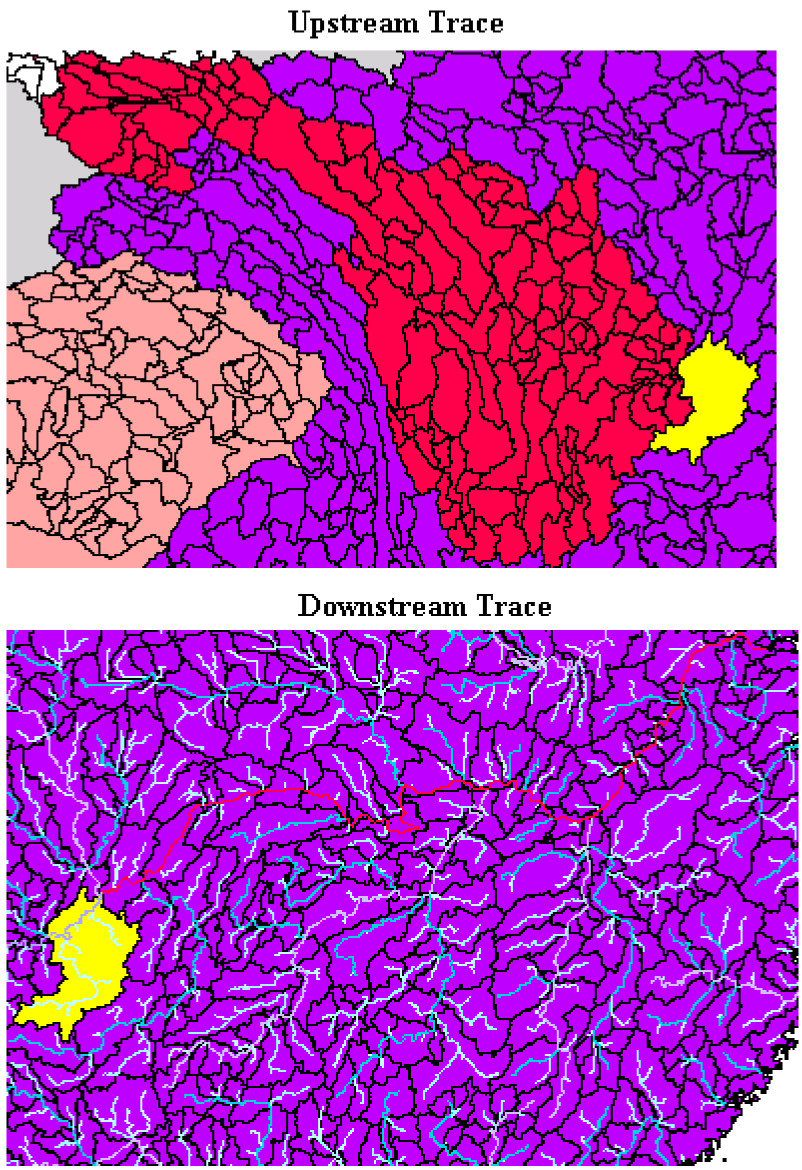


Figure 2-12 – Trace results from the EROS program
(Trace Results shown in red, initial area shown in yellow)

Downstream queries function irrespective of level, and the results are depicted as selections from the stream dataset from which the drainage areas were derived. This is interesting in that the trace is initiated by selecting a polygonal drainage area within a different theme from which the results are displayed.

Therefore, to use the EROS program to carry out navigations, one must have a streams layer and a drainage area layer. This is also the case when using the Network Analysis functionality in ArcGIS.

One feature of the EROS program is that it calculates the total area upstream of the area from which the upstream trace starts. This information is useful to watershed management planners or others with similar interests. It also suggests the capability for building hydrologic models that could run on the data identified by the search results. For example, it could be possible to use the search results to predict stream flow hydrographs after a rainfall event. Such a program, however, would require knowledge of the specific areas immediately upstream and downstream of a given area, instead of just all of the areas upstream of a given area. A stream flow hydrograph model needs to incorporate travel time and distance into its calculations, which requires a more detailed knowledge of drainage area topologies.

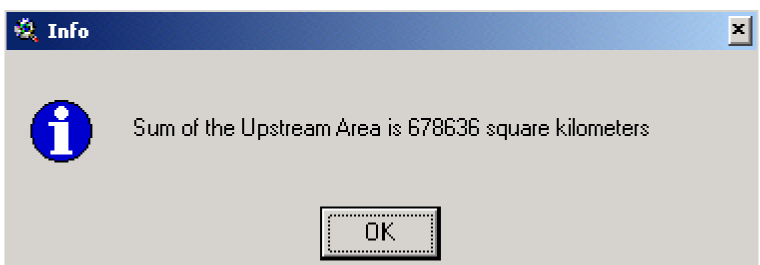


Figure 2-13 – Calculation of the Upstream Area Size with the EROS Program

The EROS program demonstrates that topology is implied by the Pfafstetter numbering system, and that this system may be used with geographic data and programs other than the ArcGIS software. However, few datasets are currently attributed with Pfafstetter codes and the system is not well known. Efforts are underway to create a more detailed watershed dataset for the continental United States. This dataset, the Elevation Derivatives for National Applications (EDNA) dataset, will contain areas attributed according to the Pfafstetter system (Franken et al, 2001). Currently, the only readily available dataset that employs the Pfafstetter system is the HYDRO1K dataset developed and disseminated by the EROS Data Center (EROS-1, 2001).

The HYDRO1K dataset is derived from the GTOPO30 digital elevation model for the globe (EROS-1, 2001). This DEM consists of approximately 1-kilometer grid cells and was derived from numerous topographic maps and surveys. The HYDRO1K dataset was created by applying the watershed delineation techniques discussed in Section 2.1 to the GTOPO30 DEM. The dataset contains vector drainage areas and streams, as well as the raster data from which the drainage areas and streams were derived. It is likely that researchers at the EROS Data Center used the flow direction grids and flow accumulation grids in assigning Pfafstetter codes to the streams and drainage areas. The exact Pfafstetter code assignment methodology has not been published.

Chapter 3: Topologic Navigation Methodology

Current techniques for performing upstream and downstream navigations on geographic data are all based on the topology implicit within the data. As described in Chapter 2, raster-based navigation techniques rely upon landscape topology as represented within a DEM grid. Vector techniques make use of the topology inherent in river networks, for all rivers must flow down-gradient across the land surface. Therefore the river network is a reflection of the landscape topology. This topology is readily viewable on the river network, and may be transferred to the drainage areas that surround each river segment in the network. However, the topologic relationships between drainage areas is not determinable without an analysis of the river network it contains. The Topographic Navigation technique described in this chapter permits navigation among drainage areas, without the necessity of a river network.

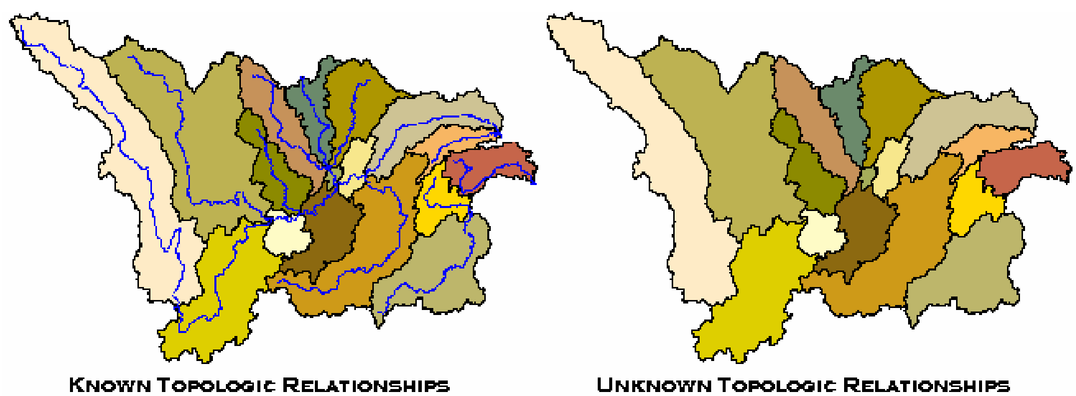
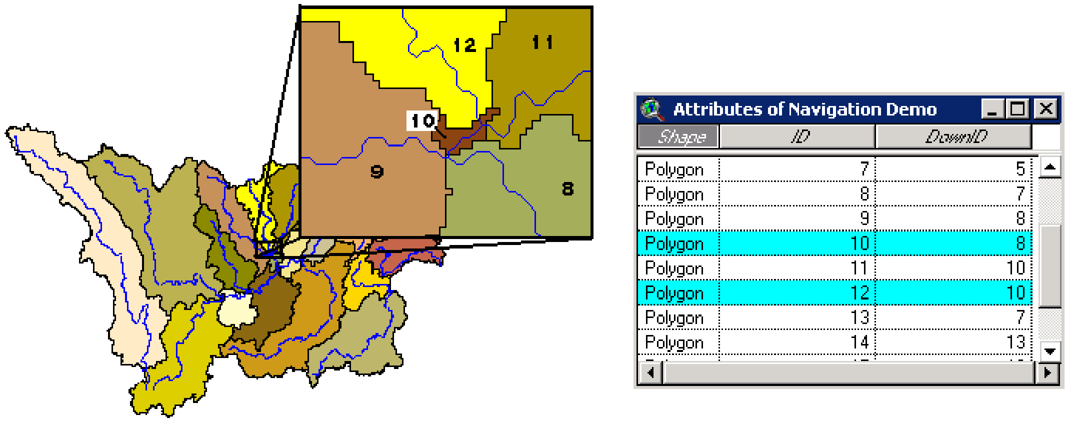


Figure 3.1 – Topologic relationships are implied by the river network

The Topologic Navigation technique derives topologic information from the required attributes of the drainage areas to which it is applied. Each drainage area must have as attributes a unique area identification code (ID) and the code of the area immediately downstream of itself (DownID). These two attributes must exist for each element in the database, and the attribute values may be determined any method (Figure 3.2).



Shape	ID	DownID
Polygon	7	5
Polygon	8	7
Polygon	9	8
Polygon	10	8
Polygon	11	10
Polygon	12	10
Polygon	13	7
Polygon	14	13

Figure 3.2 – Attributes for Topologic Navigation – Area #10 is downstream of area #12, and area #8 is downstream of #10.

The purpose of the Topologic Navigation technique is to identify the areas upstream and downstream of any given area, and to do so sequentially. The first step is to determine the areas immediately upstream each area in the database. The downstream areas are already identified through the DownID attribute. Upstream area determination is also quite simple because the ID values of the upstream areas are implied by the ID and the DownID attributes of each area in the dataset.

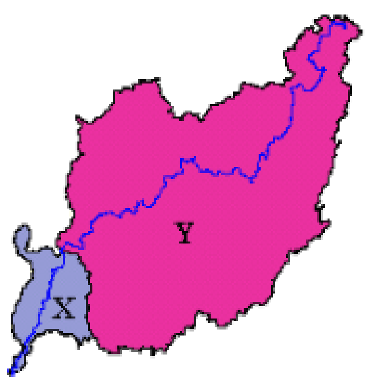
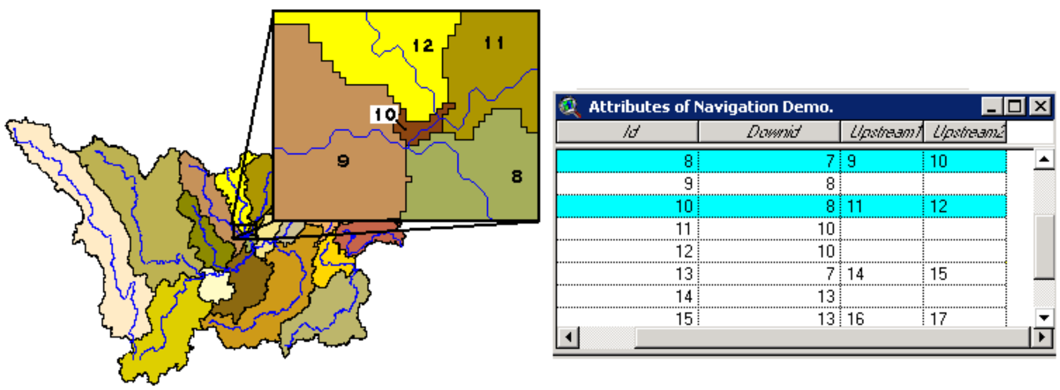


Figure 3.3 - Upstream-Downstream Relations - X is downstream of Y, therefore Y is upstream of X

If area X is immediately downstream of area Y, then area Y is obviously immediately upstream of area X. If the area immediately downstream of every individual area in the database is known, then by this process of inversion it is possible to determine the areas upstream of all areas in the database. For example, if it is necessary to determine the areas immediately upstream of area XYZ, then all that is needed is to identify those areas whose downstream area is XYZ (Figure 3.4). Depending on how drainage areas are defined, each area will have between zero and N immediately upstream areas (Figure 3.5). The ID of each upstream area must be stored as an attribute of the area for which it is upstream. These upstream area ID's are stored as "UpstreamX" attributes for each area in the dataset, where the X is an integer value ranging from 1 to N. In Figure 3.4, each area in the dataset has either 0, 1, or 2 upstream areas. In this situation, N equals 2.



Id	Downid	Upstream1	Upstream2
8	7	9	10
9	8		
10	8	11	12
11	10		
12	10		
13	7	14	15
14	13		
15	13	16	17

Figure 3.4 – Upstream Areas Determined from the DownID and ID attributes – Area 11 is upstream of area 10 because the DownID attribute for area 11 is 10

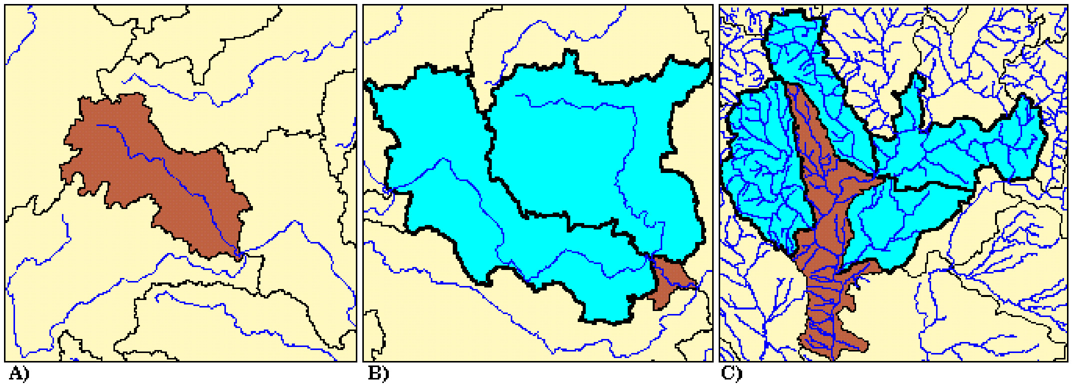


Figure 3-5 - Upstream Drainage Areas – A) Zero Areas upstream of the selected area (Brown), B) Two areas (Blue) upstream of the selected area (Brown), C) Four areas (blue) upstream of the selected area (Brown).

Once each area is attributed with the ID's of those areas immediately upstream and downstream of itself, it is possible to determine all of the areas downstream of any given area. This is achieved through the use of a "target" area, which is defined as the area on which the user is focused. The area from which the navigation is carried out is the "initial target area," and it is given a special "navigated" attribute to identify it as such. The next target area is the area immediately downstream the initial target area. This area is identified based on the DownID attribute of the initial target area. The new target area is then assigned the "navigated" attribute, and it's downstream area becomes the next target area. This process continues until an area is reached that does not have any downstream areas (Figure 3.6)

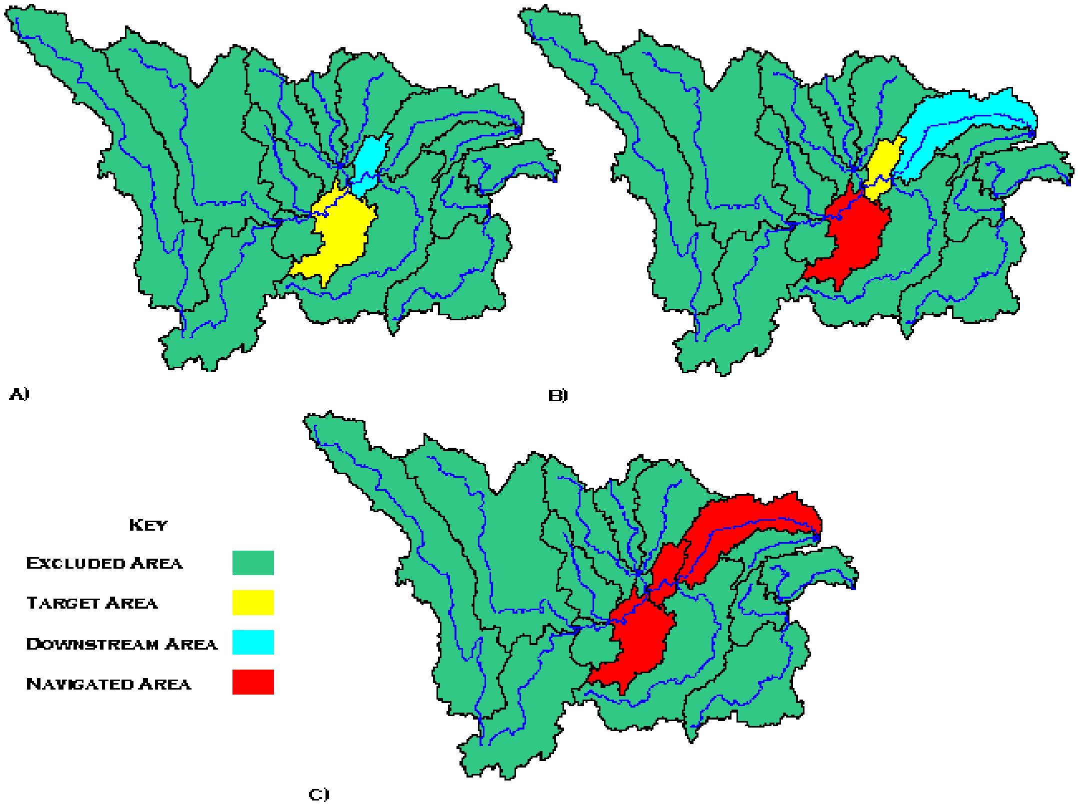


Figure 3.6 – Downstream Topographic Navigation – A) Initial target area with its downstream area, B) Previous downstream area becomes the target area, it's downstream area is recognized, the previous target area becomes a navigated area, C) Navigation ends, and all target and downstream areas become navigated areas.

After completion of the downstream navigation, the upstream navigation may commence. The upstream navigation proceeds from the initial target area, whose upstream areas are identified based on the "UpstreamX" attributes. These areas become the next target areas, and the navigation proceeds until areas are reached that do not have any further upstream areas. This procedure is analogous to the navigation procedure in the downstream direction, except that the downstream navigation only involves the identification of a single downstream area for each target area. For upstream navigations, multiple upstream areas are possible for each target area (Figure 3.7).

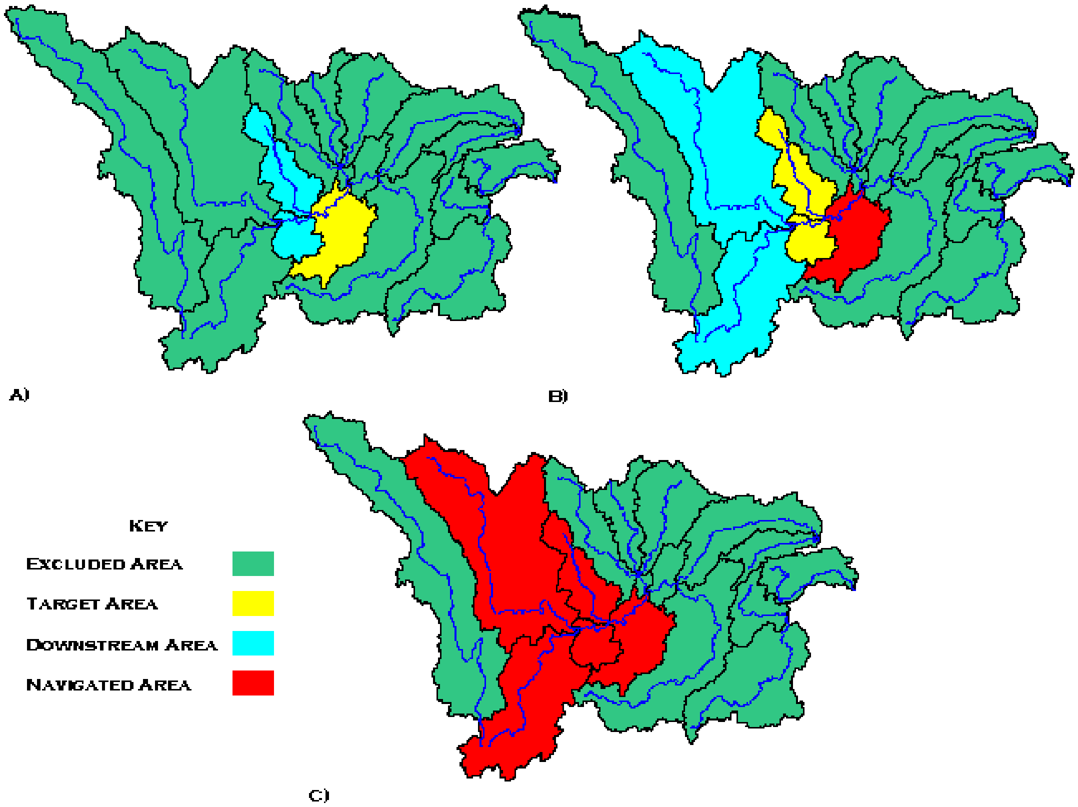
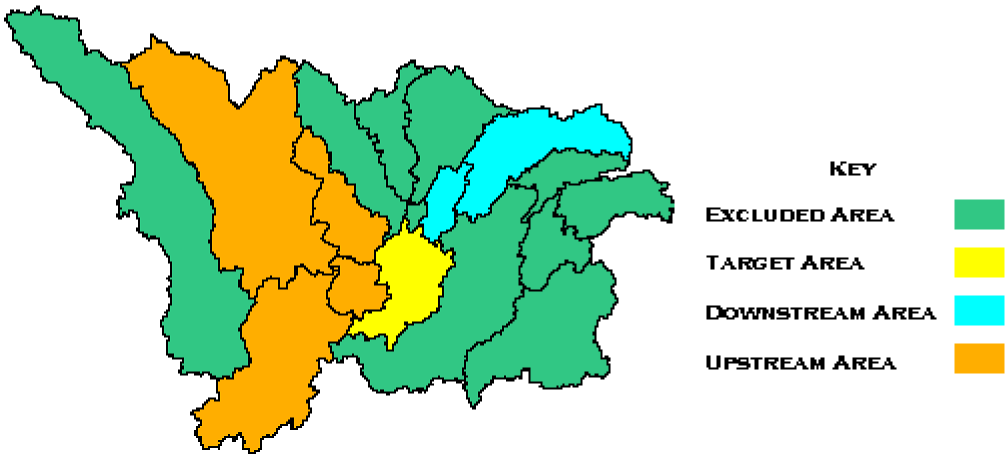


Figure 3.7 - Upstream Topographic Navigation – A) Initial target area with its upstream areas, B) Previous upstream areas become the target areas, their upstream areas are recognized, the previous target area becomes a navigated area, C) Navigation ends, and all target and downstream areas become navigated areas.

The downstream and upstream navigation results may be combined and stored as attributes of the drainage areas. This allows for multiple navigations to be performed and recorded on the same dataset. This is a unique capability of the Topographic Navigation technique with respect to the techniques discussed in Chapter 2. The implications of this results storage and other characteristics of the Topologic Navigation technique are discussed in Chapter 6.



Attributes of Navigation Demo.

Id	Downid	Upstream1	Upstream2	Nav13
5	-999	6	7	1
7	5	8	13	1
15	13	16	17	2
16	15			2
17	15	18		2
14	13			2
13	7	14	15	3

Figure 3.8 – Sample Topologic Navigation Results – the upstream areas are assigned the "2" "Nav13" attributes, with the downstream areas assigned the "1" downstream attributes. The initial target area (area 13) is assigned the "3" "Nav13" attribute, and its ID is stored as part of the Nav13 field name.

The key feature of the Topologic Navigation technique is that it derives the landscape topology based on prior knowledge of downstream areas. Therefore, in order to apply the technique, the downstream area for each entry in the database must be known. Depending on the database and how it was created, determining these downstream areas might be a difficult and time consuming task. However, if the areas have ID attributes assigned according to the Pfafstetter numbering system, the downstream areas are easily determinable. This numbering system and the methodology for downstream area determination is discussed in Chapter 4.

Chapter 4: Pfafstetter Methodology

The Pfafstetter System is a methodology for numbering drainage areas in a topographically referenced manner. Otto Pfafstetter developed the numbering system in 1989 while working as an engineer in Brazil (Pfafstetter, 1989). A strength of this numbering system is that downstream drainage areas are readily determinable based on an analysis of the drainage area numbers. The system currently is implemented in the HYDRO1K global watershed dataset (EROS-1, 2001) produced by the EROS Data Center of the U.S. Geological Survey. The following description is based upon a detailed study of the Pfafstetter system as implemented within the HYDRO1K dataset and referenced in Verdin and Verdin (1999).

It is important to distinguish between the numbering system proposed by Otto Pfafstetter (referred to as the "Pfafstetter" system) and the numbering system employed by the EROS Data Center within the HYDRO1K dataset (referred to as the "USGS-Pfafstetter" system). Subsequent sections describe the subtle differences between these two systems. Section 6.5 includes an in-depth analysis of how both of these numbering systems may be improved. A discussion of the implementation of Pfafstetter – based numbering schemes in performing topographic navigations is given in Chapter 5.

4.1 The Pfafstetter System - Definitions

In order to understand the Pfafstetter system, it is necessary to understand a few terms related to cartography. The Pfafstetter system is based upon drainage areas that are derived from linear and point features. Such features, and their definitions in the Pfafstetter system, are described in Table 4.1. Examples of each of these features are shown in Figure 4.1.

Table 4.1 - Definitions in the Pfafstetter System

River	A linear path along which water flows due to gravity
Coastline	The cartographic boundary between the land and the ocean. Water does not flow along the coastline
Drainage Area	An area that drains to a given location on a landscape. Drainage areas may contain multiple rivers.
Drainage Area Boundary:	A linear feature separating multiple drainage areas. Flow only crosses a drainage area boundary at one or two locations
Outlet	The point along the drainage area boundary through which all flow leaves the drainage area. The intersection of the drainage area boundary and the river receiving flow from the drainage area
Confluence	An outlet at which two or more rivers intersect. Only one river flows out of a confluence
Main River	The river flowing into an outlet that drains the largest area of all rivers flowing into that outlet
Tributary	The river(s) flowing into an outlet that add flow to the main river. These rivers drain areas smaller than the area drained by the main river flowing into the outlet

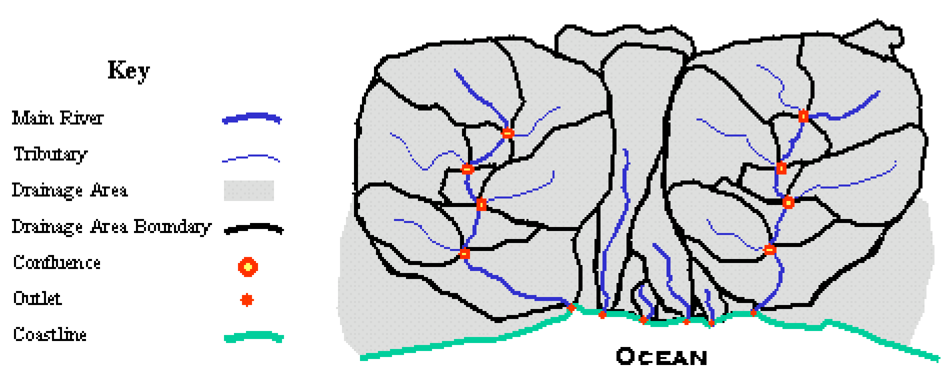


Figure 4.1 – Linear and Point Features as defined in the Pfafstetter System

Drainage areas are polygon features that are derived topographically and are based on the linear and point features described in Table 4.1. Delineation of such areas is beyond the scope of this work, although raster based delineation techniques fit well with the methodology used to implement the Pfafstetter system upon the HYDRO1K dataset. A detailed explanation of the methodology for delineating drainage areas by digital methods is given in Section 2.1.

As described in Verdin and Verdin (1999), within the Pfafstetter system there exist three types of drainage areas (Table 4.2). Examples of each of the three drainage area types are given in Figure 4.2.

Table 4.2 - Pfafstetter Drainage Areas

Basins:	The area drained by a tributary
Interbasins:	The area draining to a reach of the main river
Internal Basins:	Areas that do not contribute flow to other drainage areas or to the ocean

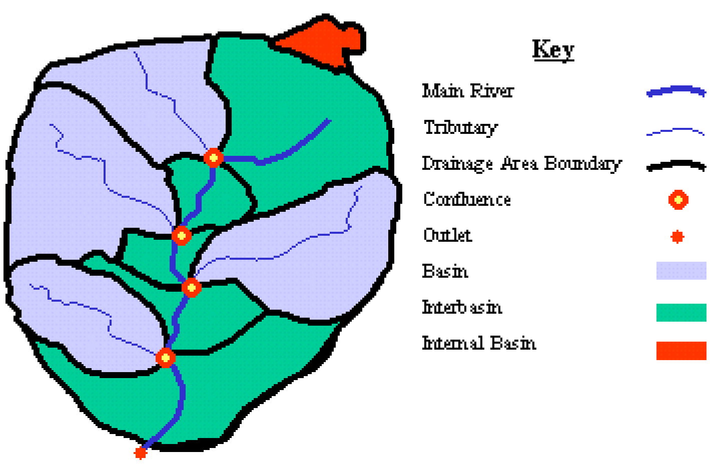


Figure 4.2 – Basins, Interbasins, and Internal Basins in the Pfafstetter System

4.2 The Pfafstetter System – Code Assignment

The Pfafstetter coding system is a hierarchical system for assigning numerical IDs to drainage areas. In this system, the land surface is divided into drainage areas at various levels, referred to as **Pfafstetter levels**. Pfafstetter Level 1 is the largest scale level, and its drainage areas cover entire continents. Pfafstetter Level 2 is the next largest scale, and the drainage areas at this scale are subdivisions of the continental scale drainage areas in Pfafstetter Level 1. Successive levels are further subdivisions of the previous level, such that level 5 drainage areas are subdivisions of level 4 drainage areas, for example. For the remainder of this discussion, the Pfafstetter levels will be referred to simply as levels, dropping the "Pfafstetter" from the title.

The Pfafstetter coding system makes use of the base 10 numbering system, which makes it easy to understand and enhances the utility of the system. Within each level, up to 10 new drainage areas are identified and numbered. For example, at level 1, up to 10 drainage areas are identified. Each of these drainage areas is assigned a 1-digit number from 0 to 9, and this number becomes the area's level 1 Pfafstetter code. The digit is referred to as the "level 1 digit." In creating the level 2 drainage areas, each of the level 1 drainage areas is subdivided into up to 10 smaller drainage areas (therefore, the maximum number of drainages at the level 2 scale is 10x10 = 100 drainage areas). Each of these new drainage areas is assigned a level 2 digit from 0-9. These assigned digits are then appended on to the end of the Pfafstetter level 1 code for the drainage area. The resulting two-digit numbers become the Pfafstetter codes for the level 2 drainage areas. These codes are of the form XY, where X and Y each may take on values from 0 to 9. The X value represents the level 1 digit and the Y value represents the level 2 subdivision of the level 1 drainage area. Higher level drainage subdivisions are numbered in the same manner, with up to ten new subdivisions for each additional level. Also, the number of digits in the Pfafstetter code of a drainage area is equal to the number of levels used to describe the drainage area. A level 6 drainage area has a 6-digit Pfafstetter code, and from this code, the drainage area's Pfafstetter codes for levels 1-5 are preserved and identifiable.

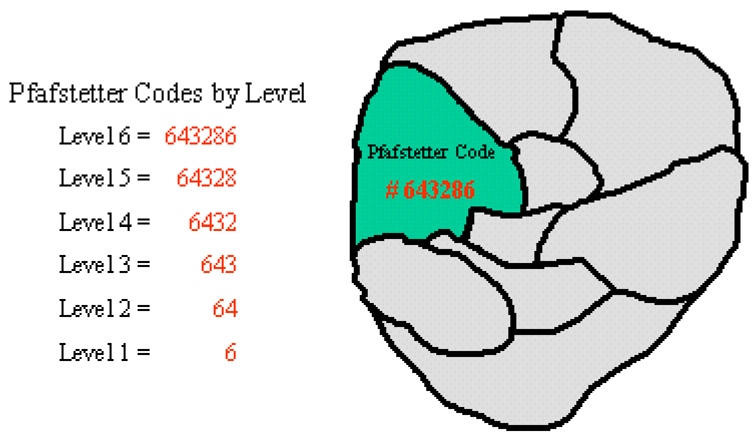


Figure 4.3 – Pfafstetter Codes by level - lower level codes are preserved at higher Pfafstetter levels

Since the lower level codes are preserved, it is possible to select groupings of drainage areas based on lower level Pfafstetter codes. For example, Figure 4.3 shows a level 6 drainage area with the ID #643286. This number is unique to the selected drainage area, and the other unidentified (gray) drainage areas also will have unique level 6 codes. However, each of these drainage areas will have the same level 5 code, #64328. This is because they all stem the same Level 5 drainage area. A query could be run on the level 6 codes to select all of the drainage areas shown in the figure. The query statement could look like:

$$643279 < \text{Level } 6 < 643290$$

Alternatively, the level 5 code for each level 6 drainage area could be stored in a separate column in the data layer's attribute table. In this way, queries may be run irrespective of levels, without the need for the mathematical reasoning used in the previous query statement. The ten drainage areas would be selected with the query:

Level 5 = 64328

The HYDRO1K data is attributed in this manner, with 6 columns in the data attribute tables storing the level 1 through level 6 codes for each drainage area. However, the use of a “search range” as demonstrated in the first query statement above is still valid, and it is extremely useful in performing navigation functions using the Pfafstetter system. This is discussed further in Sections 5.2 and 6.5.

The assignment of Pfafstetter codes (also referred to as ID numbers) is irrespective of level, and it follows the same “basic” pattern. A slight complication exists in assigning codes to areas draining directly to the ocean, but the overall procedure is unchanged. The methodology for assigning codes to ocean draining areas is discussed after the “basic” methodology for assigning codes to non-coastal areas is explained.

The code assignment procedure is based on upstream drainage areas, and may be implemented in a GIS system with either a flow direction grid or a geometric network. The assignment of ID’s is carried out by considering each individual drainage area to be subdivided. Non-coastal drainage area IDs are assigned in the following steps (refer to the definitions in Table 4.1):

- From the drainage area outlet, trace upstream along the main river, and identify the 4 tributaries with the greatest drainage area. The drainage areas containing these four tributaries are the newly identified basins.
- Assign each basin an even digit (“2,” “4,” “6,” or “8”) in the upstream direction, i.e. the most downstream basin gets the “2,” the next most downstream basin gets the “4,” etc.
- Interbasins are the drainage areas that contribute flow to the main river. The upstream and downstream boundaries of each interbasin are either a confluence between a basin tributary and the main river or the overall

drainage area outlet. Therefore, interbasins are the drainage areas between basins. Interbasins may be larger than, smaller than, or equal in size with their surrounding basins.

- Assign each interbasin the code “1,” “3,” “5,” “7,” or “9” in the upstream direction, i.e. the interbasin draining directly to the overall drainage area outlet gets the “1,” the next most downstream interbasin gets the “3,” etc. The result is that the “3” interbasin is bounded upstream by the outlet of the “4” basin, and bounded downstream by the “2” basin outlet. This pattern is continued up the main river until the “9” Interbasin is reached.
- If an area contains internal drainage areas, the largest internal drainage area is assigned the code “0” and all other internal drainage areas are incorporated into their surrounding drainage area.

The result of this numbering procedure is shown in Figure 4.4, using the drainage areas from the previous figures.

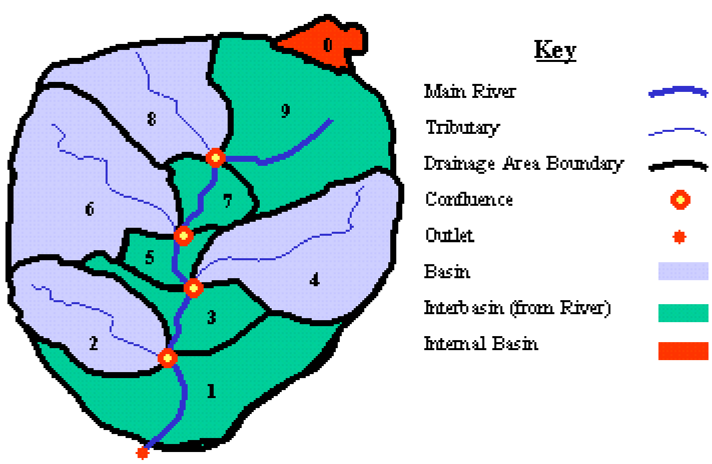


Figure 4.4 –Assigning Pfafstetter Codes for non-coastal drainage areas

The drainage areas shown in Figure 4.4 are highly idealized in that there are only 4 tributaries entering the main river along the main river’s length. This is for simplicity only. This is not a requirement of the Pfafstetter system, because only the 4 tributaries with the largest drainage areas identify basins at this level. If more than 4 tributaries are present in a drainage area, only those with the 4 largest drainage areas are considered (the other tributaries become part of an interbasin). However, if less than three tributaries are present in the drainage area, then according to the Pfafstetter system further subdivisions may not be made. This is seen in the HYDRO1K data, and may be referred to as a "drainage density" problem. This situation is discussed extensively in Section 5.2.4.

The drainage areas in Figure 4.4 may be further subdivided into the next Pfafstetter level drainage areas, as shown in Figure 4.5. The code assignment progresses in the same manner as before, however, in Figure 4.5 the tributary in basin 4 from Figure 4.4 has become the main river. Tributaries off of this new main river are used to carry out the subdivision and numbering.

As shown in Figure 4.5, the Pfafstetter codes for the higher level have the "4" as their first digit, referring back to the lower level drainage area from which the subdivisions were made. Similar higher-level areas may be delineated from the other drainage areas lower level drainage areas, as long as the lower level areas have sufficient numbers of tributaries to support higher Pfafstetter levels.

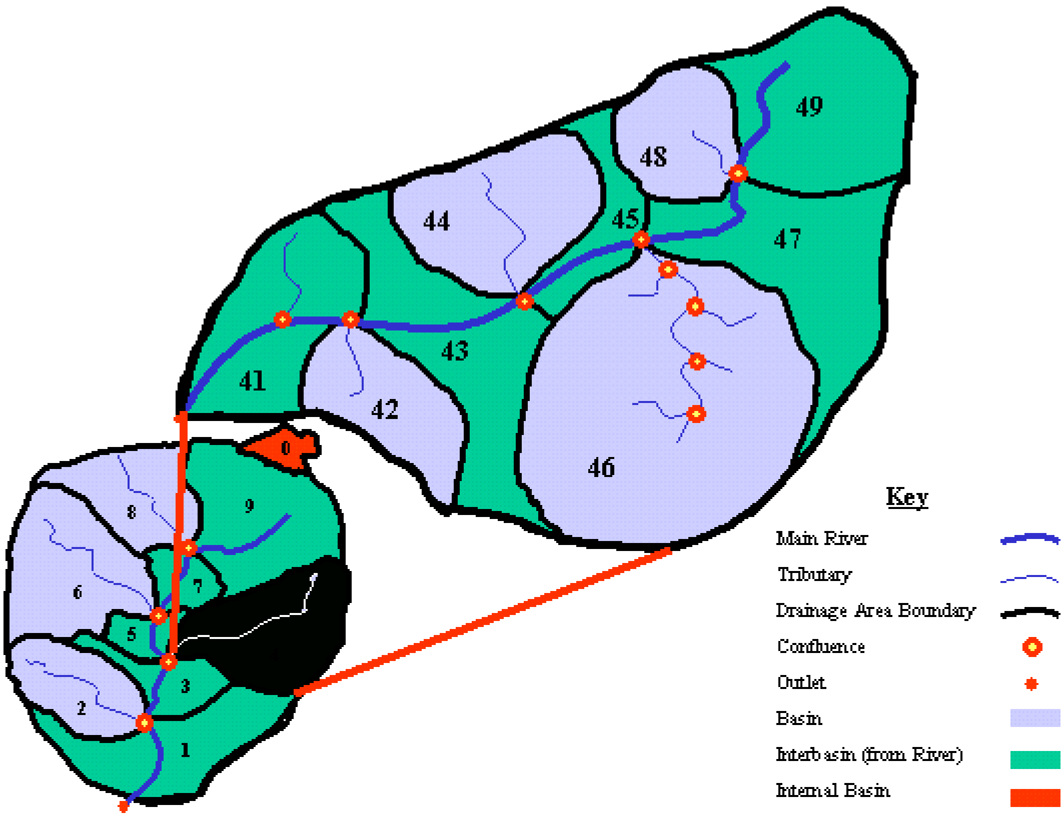


Figure 4.5 –Assigning Pfafstetter codes for non-coastal drainage areas – multiple levels

This method of assigning ID's requires that drainage areas drain to other drainage areas. This is not the case for coastal drainage areas that drain directly to the ocean. To assign ID's to these drainage areas, the procedure must be modified.

Areas draining directly to the ocean will do so either through a river or as overland flow to a coastline. An area that surrounds an ocean-discharging river is assigned Pfafstetter codes by the same procedures as discussed previously. Such an area is a coastal basin. Areas that drain to a coastline are numbered in a similar manner, however instead of tracing upstream along a main river, the trace runs along the coastline. In assigning Pfafstetter codes to coastal drainage areas, the followings steps are made:

- Identify the overall area boundary as the intersection between the coastline and an outlet whose drainage area may be numbered based on the previously discussed Pfafstetter method. For each coastal area, two such intersection points are available (Figure 3.6): the point of interest is that one point with the ocean on the left when following the coastline toward the other intersection point.
- Trace along the coastline from this intersection point, and identify the 4 rivers with the largest drainage areas that drain to the coastline. These rivers are analogous to the tributaries in the previous non-coastal area method, and their drainage areas are the new basins.
- Assign the basins the digits “2,” “4,” “6,” and “8” with the “2” basin closest to the starting intersection point.
- Those areas draining to the coastline segments in between the new basin outlets (and the intersection points) are the new interbasins. They are numbered “1,” “3,” “5,” “7,” and “9” with interbasin “1” as the area draining to the coastline segment bounded by the starting intersection point and the basin “2” outlet. The “3” interbasin drains to the next coastline segment, bounded by the “2” basin outlet and the “4” basin outlet. This process continues for each of the other new interbasins, and it ends with interbasin “9” as the area draining to the coastline segment bounded by the basin “8” outlet and the final intersection point.

This numbering procedure is equivalent to that used on interior areas if the coastline is classified as a main river. The distinction between the two methods is physical in that no overland flow is transmitted along the coastline, making the terms “upstream” and “downstream” misleading for coastal basins. Figure 4.6 demonstrates the results of this coastline procedure.

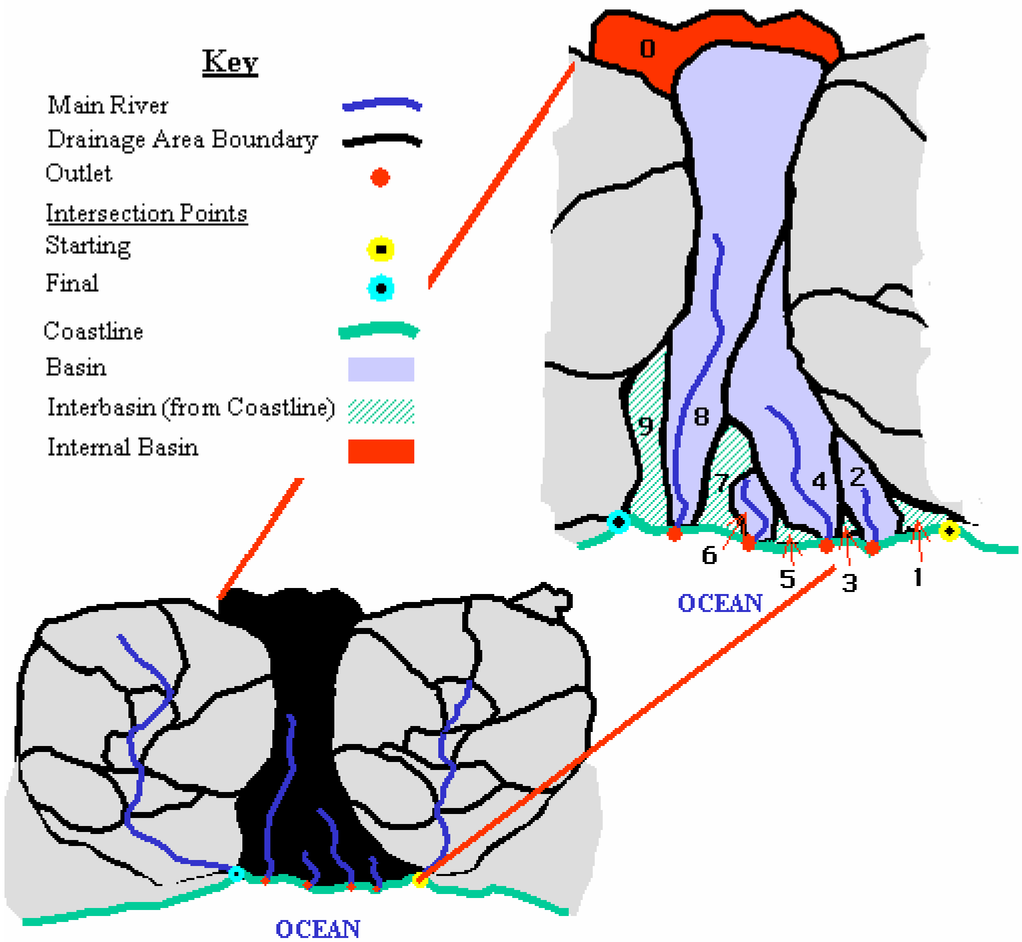


Figure 4.6 – Assigning Pfafstetter codes for coastal drainage areas

At the continental scale (Pfafstetter level 1), all drainage areas flow directly to the ocean, and as such the Level 1 Pfafstetter codes are assigned by using the coastline as the main river upon which the trace occurs. However, at the continental scale there is not a single "correct" intersection point from which to start the trace. In this situation, the four rivers with the largest drainage areas are selected based on a complete circumnavigation of the coastline. Of the selected four drainage areas, the one with the most northerly outlet is given the "2" code. The remaining basins are numbered in a clockwise fashion. The "4" area drains to the next basin outlet along the coastline from the "2" outlet when following the coastline with the ocean on the left. Level 1 interbasins are numbered based on their locations along the coastline, draining to coastline segments bounded by basin outlets. The area draining to the coastline segment bounded by the basin “8” outlet and the basin “2” outlet is divided into interbasins “1” and “9,” with the “9” interbasin bordering the “8” basin. The boundary between interbasin “1” and interbasin “9” is arbitrarily defined, and may be based on a known geographic feature.

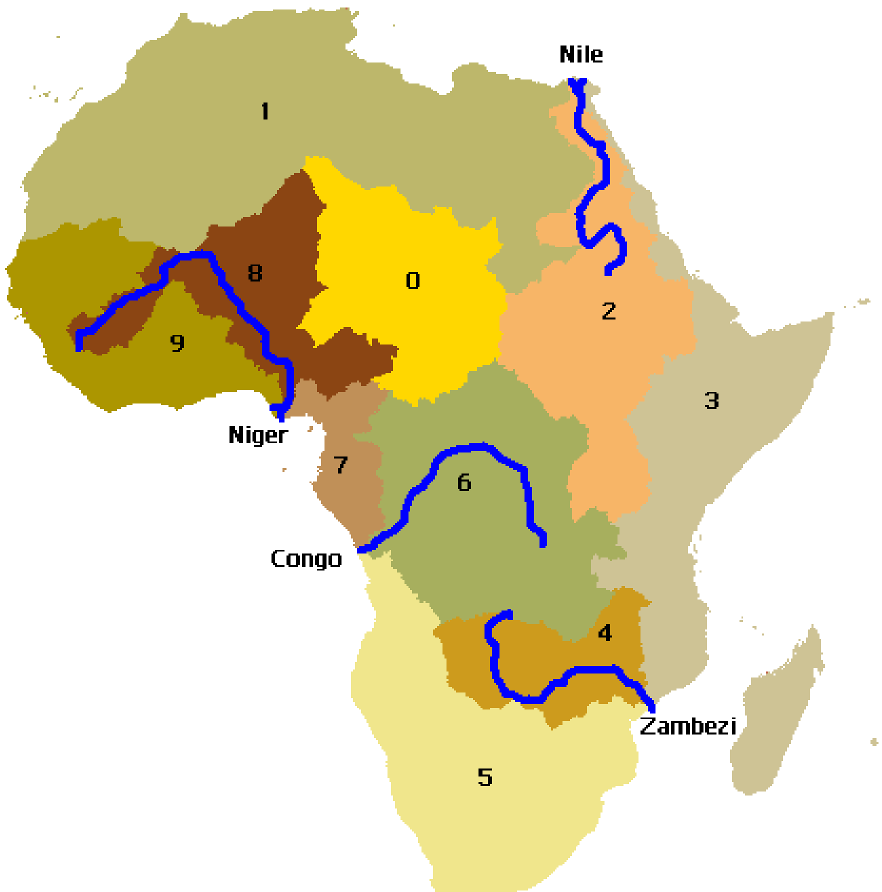


Figure 4.7 – Level 1 Pfafstetter Codes for Africa (HYDRO1K)

Figure 4.7 shows the level 1 drainage areas for Africa. Here the basins contribute flow to the Nile River, the Zambezi River, the Congo River, and the Niger River rivers, with basin “2” defined as the Nile River drainage area because it has the most northerly outlet. Note that the boundary between interbasins “1” and

“9” is arbitrarily defined. It coincides with the drainage area boundaries of higher Pfafstetter levels.

In considering all of the methods and procedures discussed previously, it is clear that the drainage areas shown in Figure 4.1 may support 2 levels of Pfafstetter codes. Note that because these watersheds drain only to a section of a coastline, it is not possible to assign unique codes to these drainage areas (Figure 4.8). There are multiple sets of codes that would be acceptable and accurate in the Pfafstetter system. This is because it is impossible to determine to which specific level 1 area each of the areas in Figure 4.1 contribute flow. It is also possible that the drainage areas shown are not level 2 areas, but rather higher level areas. However, the spatial inter-relationships among the drainage areas, as implied by the Pfafstetter codes, are identical regardless of the actual level or code set used.

The key point in Figure 4.7 is that the actual code assigned to a drainage area is set by convention. It is by convention that the “2” code is assigned to the most northern level 1 basin. It is also suitable to assign the “2” code to the most southern basin, or to the basin with the largest drainage area. Also, the “direction” of the trace along the coastline (i.e. with the ocean to the left) was also set by convention; progressing with the ocean to the right is also theoretically valid. The topographical relationships between drainage areas, however, are not affected by these arbitrary decisions, and therefore are irrespective of the code set employed. These topographical relationships are inherent within the Pfafstetter methodology for drainage area numbering, and it is inherent in whichever numbering set is used to describe an area. The Pfafstetter codes are assigned based on the relative topology of the drainage areas, and by analyzing assigned codes, this relative topology is discernible. With the Pfafstetter system, it is possible to determine each and all drainage areas upstream and downstream of any given area. This characteristic allows for topographic navigation through the landscape, with only an inspection of the Pfafstetter codes.

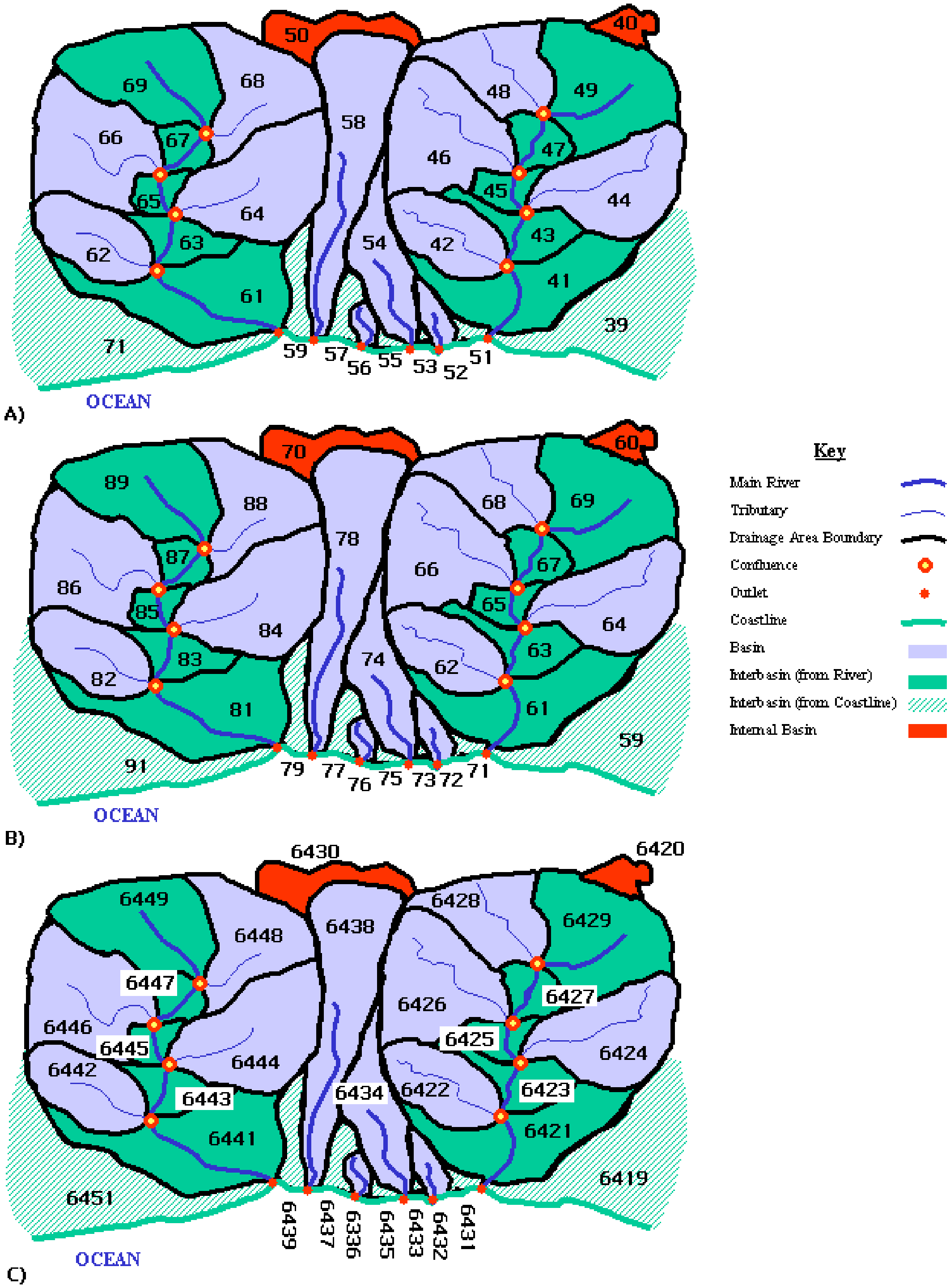


Figure 4.8 – Alternative Sets of Acceptable Pfafstetter Codes – A) and B) are level 2 codes, C) possible level 4 codes. The topological relationships between the drainage areas is maintained regardless of the code set (A, B, or C)

4.3 Downstream Area Identification Using Pfafstetter Codes

4.3.1 Introduction

The strength of the Pfafstetter system of area codification is that the system assigns numerical IDs based on topology. It is therefore possible to determine basins that are downstream of a given basin based solely on tabular data that includes the numerical Pfafstetter codes. For example, interbasin 843 is downstream of interbasin 845, and drains to interbasin 841. In this discussion, the term "downstream" refers to the single area immediately downstream of a given area, and not all of the areas downstream of that area. The entire set of areas downstream of a given area is determinable with the Topographic Navigation technique described in Section 3.2.

4.3.2 Topologic Properties of Drainage Area Classifications

Pfafstetter codes are assigned by progressing upstream along main rivers. All upstream areas have integer codes greater than the codes of the downstream areas, and the difference between the codes of two areas on the same river system is directly proportional to the number of drainage areas separating the two drainage areas within the system. Therefore, the area immediately downstream of any given area has a code nearly identical to but slightly smaller than the code of the area itself. The actual numerical difference between codes of a given area and its downstream drainage area depend upon the Pfafstetter classification of the given area type.

By definition, basins are drainage areas that drain to tributaries. For such drainage areas, water flows across the drainage area boundary only at the outlet point. Therefore, these areas do not have any upstream drainage areas. This characteristic prevents basins from being areas downstream of any other areas. However, because basins are defined based on tributaries and have outlets corresponding to main river confluences, each basin must drain into a drainage area that contributes flow to a main river segment. Thus, all basins must drain into interbasins. The only exceptions to this rule are the instances where basins drain directly into the ocean, such as at the level 1 scale (Figure 4.9).

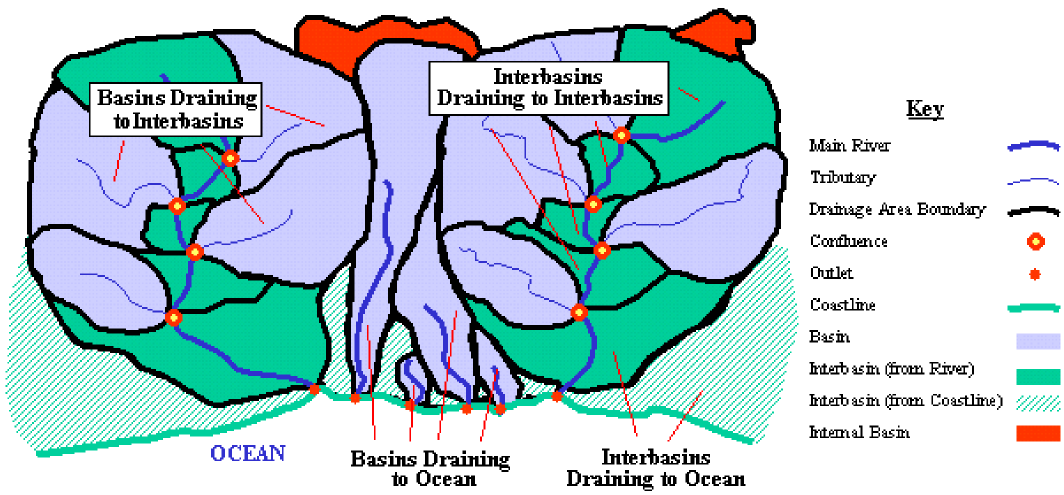


Figure 4.9 – Drainage Characteristics of Pfafstetter Area Classifications

Non-coastal interbasins must also drain to interbasins. Each interbasin is bounded at the downstream end by a confluence point between a tributary and the main river. Flowing downstream from the confluence is a main river segment, which receives flow from the surrounding interbasin. Similarly, each interbasin is bounded at the upstream end by a confluence point at which flow from at least one basin and an interbasin merge. Therefore non-coastal interbasins may be downstream of either basins or other interbasins. Coastal interbasins, however, do not always share the same upstream-downstream properties of non-coastal interbasins. Coastal interbasins drain directly to the ocean and do not have any downstream drainage areas. Yet these coastal interbasins may or may not have upstream areas. The existence of upstream areas depends on the lower level areas containing the coastal interbasin. This ambiguity in upstream and downstream relations between coastal and non-coastal interbasins exists at all Pfafstetter scales, and it must be resolved in order to correctly determine topologic relations between Pfafstetter drainage areas. Specifically, a distinction must be made between coastal and non-coastal interbasins.

The third type of Pfafstetter drainage area is the internal basin. This area does not contribute flow to other areas or to the ocean. Therefore, these areas will not have any downstream areas. Under the USGS-Pfafstetter system, these areas will also not have any upstream drainage areas. In this system, internal basins are completely unrelated to the drainage areas around them. However, in the applying the strict internal basin definition as employed in the Pfafstetter system (Pfafstetter, 1989, Verdin 1999), situations may exist in which internal basins may be downstream of other drainage areas. This special scenario is discussed in Section 5.2.4, and is not addressed in the following discussion.

4.3.3 Determining Downstream Areas – Numerical Analysis

Upon excluding the special cases involving interbasins, it is accurate to state that the areas downstream of non-coastal basins and interbasins must themselves be interbasins. These downstream interbasins will have odd-numbered codes equal to the code of the upstream areas, minus some "detraction."

$$\text{Downstream Code} = \text{Upstream Code} - \text{Detraction} \qquad \text{Eqn. 4-1}$$

The value of the "detraction" term depends on the value of the highest-level digit or digits in the upstream area's code. In the Pfafstetter system, this highest level digit for interbasins and basins may take on any value in the range {1…9}. For the purposes of determining the codes of the downstream areas, it is useful to divide the nine digits in this range into three separate categories, for codes that have highest level digits in these categories have equivalent "detraction" terms. The three categories are given in Table 4.3, and the development of the detraction term for each category is discussed in a following sub-section.

Table 4.3 - Highest-Level Digit Categories for Determining Downstream Elements

	Name	Possible Highest Level Digits
Category A	"Even"	{2, 4, 6, 8}
Category B	"Odd not One"	{3, 5, 7, 9}
Category C	"One"	{1}

Category A – The "Even" Highest Level Digits

A drainage area with a code in this category must be a basin, and it will drain to the interbasin with the highest number lower than its own code. This will always be the code minus one, so the detraction term for this category is "1." In order for a given basin number to exist in the database, there must also exist an interbasin with the code one less than the code of the basin. In assigning Pfafstetter codes, the upper boundary of the downstream interbasin is denoted by the confluence of the tributary that receives drainage from the most downstream basin. This basin is given the "2" digit to signify that it is the second most downstream area within the lower-level area, and it must drain to the first downstream area, specifically the interbasin with the "1" highest-level digit. Upstream of the confluence, the area draining to the main river is assigned the "3" highest-level digit, and the "4" basin drains to the tributary whose confluence with the main stem marks the upstream boundary for the "3" interbasin. This pattern continues for all of the digits, without exception (Figure 4.10).

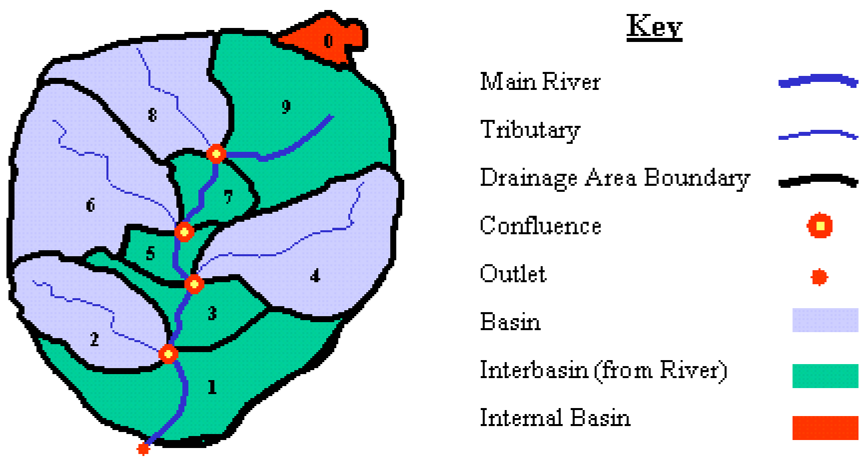


Figure 4.10 – Category A for Downstream Determination - Basins always drain to interbasins with the code one less than their own. Interbasins without the highest level digit "1" always drain to interbasins with the code two less than their own.

Category B – The "Odd not One" Highest Level Digits

Drainage areas with codes falling in this category are interbasins that do not drain to the outlet of a lower-level area. These areas drain to an interbasin with the next highest interbasin code. This is always the interbasin with the code two less than the code of the interbasin in question, and the detraction term for this category is "2." For example, interbasin 25 will drain to interbasin 23, and interbasin 23 will drain to interbasin 21. In assigning Pfafstetter codes, the upper boundary of the downstream interbasin (area #1) is denoted by the confluence of the tributary that receives drainage from the most downstream basin (area #2). This confluence also marks the downstream boundary of the area #3 interbasin, which obviously contributes flow to the "1" interbasin downstream. The upstream boundary of area #3 is also identified by a confluence that receives flow from area #4 (a basin) and area #5 (an interbasin). This pattern continues for all of the digits, without exception (Figure 4.10).

Category C – The "One" Highest Level Digits

An interbasin with a code ending in "1" poses a unique difficulty, for the other digits in the code must also be considered when determining the downstream element. It is not sufficient to use the "even/odd" characteristic of the highest-level digit in order to determine downstream drainage. It would be incorrect to assert that interbasin 231 drains to interbasin 229, as would be predicted if Category B were to include areas with the highest level digit "1." This scenario would suggest that at the level 2 scale, interbasin 23 drains to basin 22, which does not fit the Pfafstetter numbering system (basins do not receive flow from other areas).

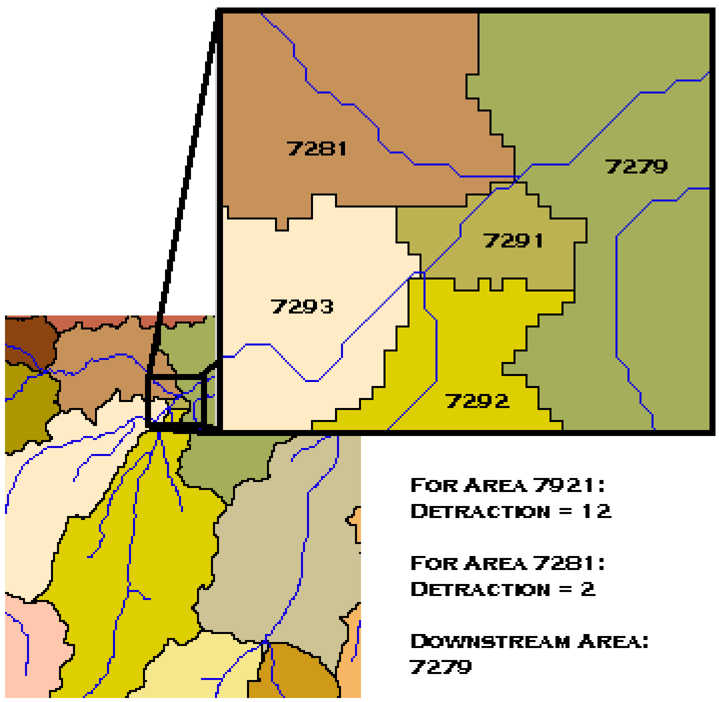


Figure 4.11 – Category C for Downstream Determination – Area 7297 is downstream of Category C areas #7281, #7291

According to the Pfafstetter system, interbasin 7291 drains to interbasin 7279 (Figure 4.11). To understand why this is so, it is important to recognize three facts:

1. Interbasin 729 must drain to interbasin 727
2. Interbasin 7291 is the area most downstream of the level 4 drainage areas within the level 3 area #729.
3. Interbasin 7279 is the area most upstream of the level 4 drainage areas within the level 3 area #727.

With these facts, it is evident that any flow leaving interbasin 7291 must immediately enter the area downstream of the level 3 area #729. With the level 3 area #729 falling into Category B, this downstream level 3 area is area #727. Also, any flow entering the level 3 area #727 must first enter the level 4 area #7279. Therefore, at the level 4 scale, it is clear that area #7291 must drain into area #7279. For this scenario, the detraction term is "12."

The above analysis demonstrates that for codes falling into Category C, it is not sufficient to only consider the highest-level digit when developing the detraction term. It is necessary to consider the topographic relationships of the lower level drainage areas that contain the area falling into Category C. In the previous example for the level 4 area #7291, it was necessary to consider the topographic relationships inherent in the level 3 area to which area #7291 belongs. This lower level area #729 is an interbasin whose downstream area is determined based on the detraction term from Category B. However, the lower level area could have a code pertaining to either Category A, B, or C. Each of these possibilities will yield a different value for the detraction term.

For example, the area downstream of area #7281 is also area #7279 (Figure 4.11). In this case, the detraction term is "2," and the level 3 area #728 falls into Category A. At the level 3 scale, area #728 drains into area #727. Following the logic from the previous example, at the level 4 scale the most downstream area within level 3 area #728 must drain into the most upstream area within the level 3 area #727. Therefore area #7281 must flow into area #7279.

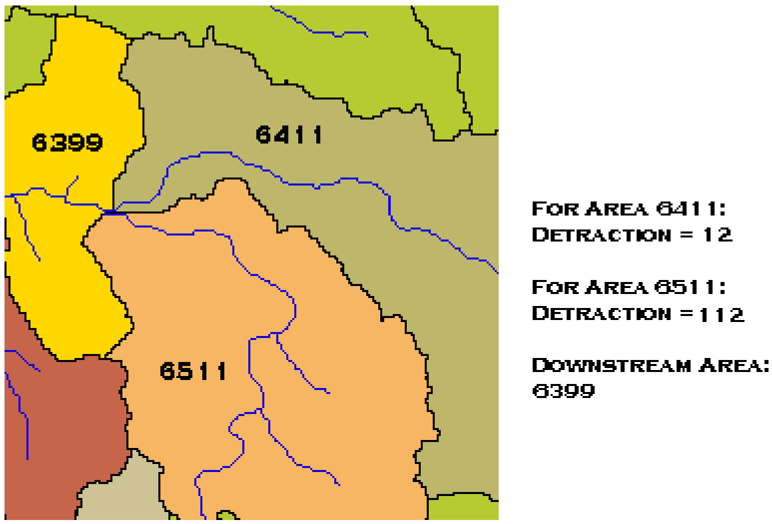


Figure 4.12 – Downstream area determination for lower level Category C areas

The situation becomes more complex if the lower level area also falls into Category C, as is the case for the level 4 area #6411 (Figure 4.12). It is now necessary to consider the 3rd highest level digit, because this is the highest level digit that does not fall into Category C. This digit is "4," which corresponds to the level 2 drainage area #64. This basin falls into Category A, and it must drain to the level 2 area #63. At the level 3 scale, this level 2 area is divided into 9 sub-areas, of which area #641 is the most downstream. Following the logic from the previous example, area #641 must drain into area #639, which is the most upstream level 3 area within level 2 area #63. This area #641 is further divided into level 4 sub-areas, of which area #6411 is the most downstream. It must drain directly into the most upstream level 4 sub-area within level 3 area #639, which is area #6399. Therefore area #6399 is downstream of area #6411. The detraction term in this scenario is "12."

For a final example, consider area #6511 (Figure 4.12). At the level 2 scale, area 65 drains to area #63. Therefore at the level 3 scale area #651 drains to area #639, and at the level 4 scale area #6511 drains to area #6399. The detraction term for this scenario is equal to “112.” This value is substantially larger than the other detraction terms from the previous examples, even than that for the similar level 4 area #6411. The difference in detraction terms between these two level 4 areas stems from the different categories into which their highest-level “non-one” digits fall.

From these examples, it is clear that two parameters are important in determining the value of the detraction term for an area. The first parameter is the category of the highest-level “non-One” digit in the area’s code. The second parameter is the number of consecutive “1” digits in the code, starting from the highest digit. These two parameters may be combined into equations that are used to predict the detraction term and therefore the value of the downstream area’s code. For description purposes, the second parameter is named the variable “K.” The appropriate equations are given in Table 4.4, with the variable “K” reflecting the second parameter.

Table 4.4 - Detraction Equations

Parameter 1	Equation	
Category A	$Detraction = 1 + Z \cdot \sum_{i=0}^{K-1} 10^i \rightarrow Z = \begin{Bmatrix} 1 \text{ for K} \geq 1 \\ 0 \text{ for K} \leq 0 \end{Bmatrix}$	Eqn. 4-2
Category B	$Detraction = 1 + \sum_{i=0}^{K} 10^i$	Eqn. 4-3

These detraction equations make use of the base-10 characteristics of the Pfafstetter system, as well as the even-odd numbering characteristics of basins and interbasins. The results of the equations applied to each of the four previous examples are given in Table 4.5. As shown, the detractions calculated with the equations in Table 4.4 match those from all of the previous examples. Through the use of the detraction equations in combination with Eqn. 4.1, the areas downstream of all Pfafstetter-attributed non-coastal drainage areas may be readily determined.

Table 4.5 - Comparison Between Stated and Calculated Detraction Terms

Area	Stated Detraction	Parameter #1	Parameter #2	Equation	Equation Results	Calculated Detraction
32	1	A	0	$Detraction = 1 + 0 \cdot \sum_{i=0}^{-1} 10^i$	1	1
25	2	B	0	$Detraction = 1 + \sum_{i=0}^{0} 10^i$	1+1	2
7291	12	B	1	$Detraction = 1 + \sum_{i=0}^{1} 10^i$	1+1+10	12
7281	2	A	1	$Detraction = 1 + 1 \cdot \sum_{i=0}^{0} 10^i$	1+1	2
6411	12	A	2	$Detraction = 1 + \sum_{i=0}^{1} 10^i$	1+1+10	12
6511	112	B	2	$Detraction = 1 + \sum_{i=0}^{2} 10^i$	1+1+10+100	112

4.3.4 Determining Coastal Areas – Numerical Analysis

According to the previous section, it is possible to determine the area downstream of each non-coastal area in a Pfafstetter-attributed dataset. Only interbasins and basins away from the coast will have downstream areas. All areas along the coastline will drain directly to the ocean and not to any other drainage area in the dataset. Therefore, it is necessary to distinguish between coastal and non-coastal areas when determining downstream areas.

In distinguishing between coastal and non-coastal areas, it is not necessary to consult maps of refer to any form of geographic information. Coastal areas are easily identifiable through an analysis of the digits in an area’s Pfafstetter code. Specifically, all coastal areas have codes that fit one of three numerical criteria.

These criteria are listed in Table 4.6 and are individually discussed in the following subsections.

Table 4.6 – Ocean Draining Criteria for Pfafstetter-Attributed Drainage Areas

	Description	Example
Criterion #1	All odd digits, except the possibly the highest level digit	159
Criterion #2	An even level 1 digit, all the rest of the digits are “1”	811
Criterion #3	Odd digits, then a single even digit, and all “1” digits	161

Criterion #1 – All digits, except possibly the last digit, are odd

This criterion stems from the method of assigning Pfafstetter codes at the lower levels, beginning with the level 1 continental scale areas. At level 1, all areas drain to the ocean (excluding the possible internal basin). The interbasins, which have odd Pfafstetter codes, drain directly to the ocean through a coastline. These areas are bounded by two basins, with the one exception being the boundary between area “1” and area “9” which is arbitrarily defined (See Section 4.2). The even numbered areas, which are the level 1 basins, also drain directly to the ocean, but unlike the interbasins, they do so through a river. In considering a level 1 interbasin divided into level 2 areas, it is possible to see a pattern in the assigned codes of ocean draining areas.

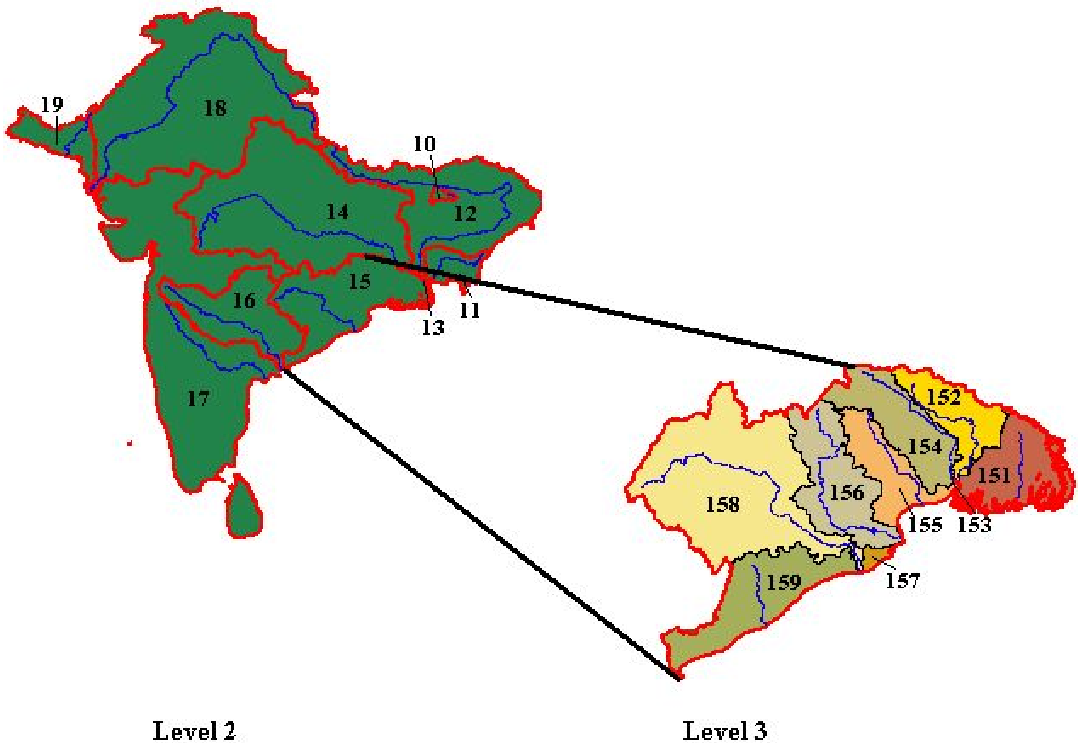


Figure 4.13 – Areas Draining to the Ocean – Criterion #1, Levels 2 and 3 from a level 1 interbasin

The four largest river basins draining to the ocean within each level 1 interbasin are given an even level 2 digit. The remaining areas between these basins, which also drain to the ocean, have odd level 2 digits and are interbasins. Thus the level 2 areas have codes of the form XY, where X is an odd number, and Y is either even or odd. In the level 3 delineations from the level 2 interbasins, all of the new areas must also drain to the ocean through the coastline. Again, the 4 largest river basins draining directly to the coast are assigned even level 3 digits, and the remaining areas are given odd digits. These areas all have codes of the form XYZ where X is an odd number, Y is an odd number, and Z is either even or odd. In comparing these codes from the level 2 and level 3 analyses, it is clear that both codes contain odd numbers for every digit other than the highest-level digit. This pattern continues for every successive level in the Pfafstetter hierarchary as long as higher-level areas are created from interbasins. Therefore interbasins draining directly to the coastline have odd values for each digit, and coastal basins have all digits odd except the highest-level digit, which is even. Most of the coastal areas have codes that meet this criterion.

Criterion #2 – The level 1 digit is even, and the remaining digits are 1

The second criterion for ocean drainage considers interbasins that drain to main river segments that intersect the coastline. Specifically, the criterion focuses on the even numbered areas (basins) at the level 1 scale. In dividing such basins into level 2 basins, the created X1 interbasin must drain to the ocean. Water from all other basins and interbasins (X2…X9) in the level 1 basin must first pass through interbasin X1 before reaching the ocean. In progressing to a level 3 delineation, level 2 interbasin X1 is divided into areas X1Y where Y may take on values from 0 to 9. The most downstream area is therefore X11, and it must drain to the ocean. Water from all of the other X1Y areas must flow through area X11 before reaching the ocean. This pattern continues for all subsequent higher-level divisions from the initial level 1 basin. Incidentally, criteria 2 considers only the case of even level 1 basins because the code pattern X1, X11, X111 etc. with X as an odd number is considered in Criterion 1. Criterion #2 is applied only four times on each continental landmass, for it applies only to the most downstream area within each of the four level 1 basins.

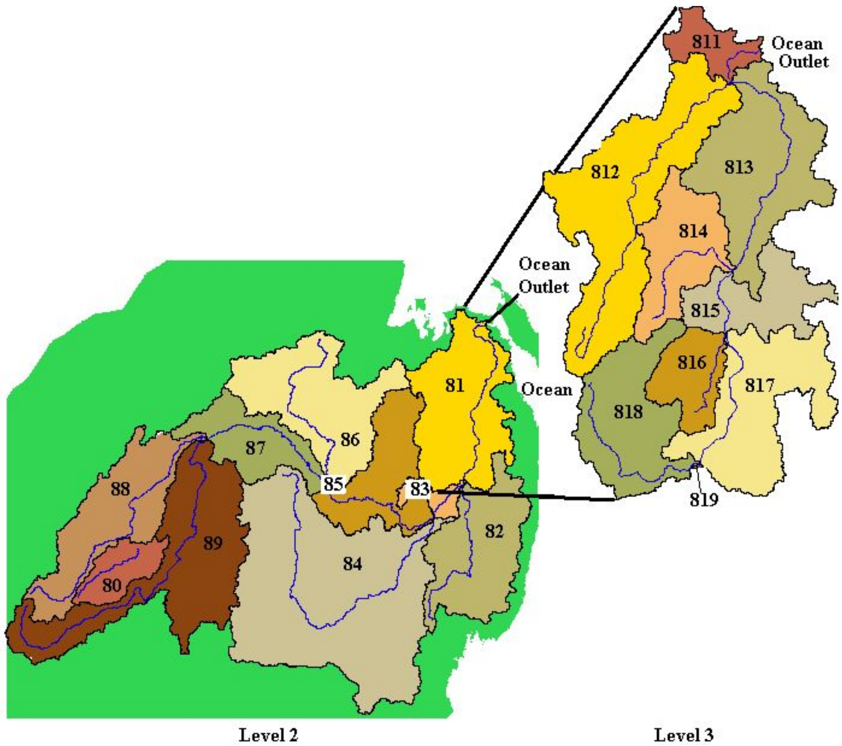


Figure 4.14 – Areas Draining to the Ocean – Criterion #2, Levels 2 and 3 from a level 1 basin

Criterion #3 – Odd digits, then a single even digit, and all "1" digits

The final criterion is really a combination of the previous two criteria. It states that if the first few of digits are odd, then there is a single even digit and all the remaining digits are "1," the area drains to the ocean. As in Criterion #2, such an area is an interbasin draining to a main river segment that intersects the coastline.

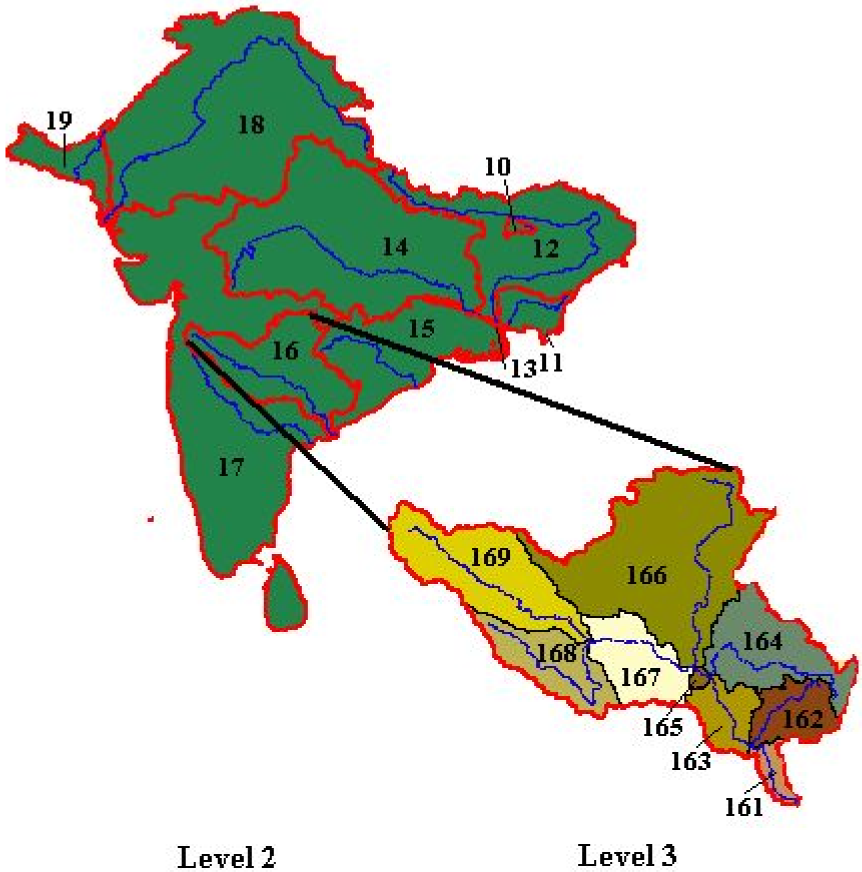


Figure 4.15 – Areas Draining to the Ocean – Criterion #3, Levels 2 from a level 1 interbasin, level 3 from a level 2 basin

As already discussed in connection with Criterion #1, all level 1 areas drain to the ocean, regardless of whether they are basins or interbasins. If level 2 areas are derived from a level 1 interbasin, then all of those areas also must drain to the ocean. This third criterion considers the case where the level 3 areas are derived from a level 2 basin that meets criterion #1. For these level 3 areas, only the most downstream area drains to the ocean. This area will have the Pfafstetter code XY1, where X is an odd number and Y is an even number. Higher level codes will fit this criterion in various ways, and sample qualifying code forms are given in Table 4.7.

Table 4.7 - Pfafstetter Code forms meeting Criterion #3 for Coastal Areas

KEY	Level 3	Level 4	Level 5	Level 6
	XY1	XXY1	XXXY1	XXXXY1
X = Odd Number		XY11	XXY11	XXXY11
Y = Even Number			XY111	XXY111
1 = One				XY1111

The important characteristic of all of these code forms are that the lower level digits are odd, then the next level digit is even, and then all higher level digits are "1." Interbasin 161 fits this pattern, and as shown in Figure 4.15, it drains directly to the ocean. More areas have codes with this criterion than those with codes meeting criterion #2. However, criterion #3 is met less often than criterion #1.

4.3.5 Determining Internal Basins – Numerical Analysis

The only drainage areas yet to be discussed are the internal basins. These areas are the simplest areas to identify, because they are the only areas with a "0" highest-level digit. Such areas do not contribute flow to any other areas, and therefore do not have any downstream areas.

4.3.6 Downstream Area Identification - Summary

For datasets attributed according to the Pfafstetter system, the downstream drainage area is easily identifiable through an analysis of the Pfafstetter code. Downstream areas are determinable through the application of simple equations that may be implemented into a computer program. It is also possible to determine those areas that drain to the ocean. Such coastal areas have Pfafstetter codes that satisfy one of three numerical criteria. These criteria stem from the methodology involved in assigning Pfafstetter codes. Finally, internal basins are easily identified as areas with "0" as their highest-level digit.

The topology inherent in the Pfafstetter system numbering allows for the automatic determination of downstream elements. Therefore, the Topologic Navigation technique (Chapter 3) could be readily applied to datasets attributed according to the Pfafstetter system because the downstream elements are easily discernible.

Chapter 5: Implementation on HYDRO1K Data

In order to determine their applicability and effectiveness, the Topologic Navigation methodology (Chapter 3) and the Pfafstetter-based Downstream Area Identification methodology (Section 4.3) were incorporated into a suite of macros within the Microsoft Excel spreadsheet program. These macros, collectively named the "Pfafstetter Tools" carry out three separate functions:

- Downstream Area Identification
- Upstream Area Identification
- Topologic Navigation

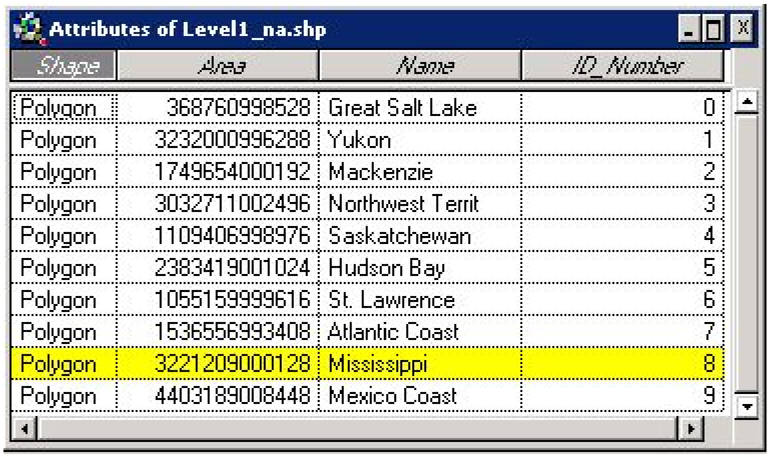
The macros were developed in order to perform Topologic Navigations on HYDRO1K data produced by the EROS data Center of the USGS (EROS-1, 2001). As discussed in Chapter 2, the HYDRO1K dataset is a global database consisting of raster-delineated drainage areas. These areas have ID values assigned based upon the Pfafstetter numbering system as described by Verdin and Verdin (1999). As mentioned in Chapter 4, subtle differences exist between the Pfafstetter numbering methodology and the methodology used in attributing the HYDRO1K data. These differences are addressed in this chapter, and for this discussion the methodology for assigning codes to HYDRO1K data is referred to as the "USGS-Pfafstetter" system.

The macros described in this chapter were developed in order to run on hydrologic data from various sources, including drainage area datasets with attributes assigned according to either the Pfafstetter system or the USGS-Pfafstetter system. As a result, the macros incorporate the differences between the two Pfafstetter-based numbering systems, and these differences are described in the following discussion of the Downstream Area Identification macro. This macro carries out the processes discussed in Section 4.3 and includes the necessary modifications for use on the USGS-Pfafstetter attributed data. The Upstream Area Identification macro and the Topologic Navigation macro carry out the processes described in Chapter 3. With slight modifications, these macros will work in any other COM compliant database management software. Sections 5.2-5.4 describe each of the Pfafstetter tools macros, and Section 5.1 describes the specific programming concepts employed within the macros. The procedure for viewing the data in the ArcGIS or ArcView software packages is also described.

5.1 The Pfafstetter Tools – Programming Concepts

5.1.1 – Data Characteristics and Tabular Database Concepts

The Pfafstetter Tools are designed for use on geographic data with editable attributes. This data may be stored as an ArcView shapefile, an ArcGIS geodatabase, or a simple table in either software package. Each of these file types store data as attributes of a specific geographic location, and they display these attributes in a database with a tabular format. The columns in the database are referred to as "fields," whereas the rows are referred to as "records." There is one record for each distinct entry in the database, and the number of fields depends on the number of different attributes the dataset contains. Figure 5.1 shows a sample database for watersheds. As shown, there are 9 watersheds in the database (9 rows/records), and the database contains the area, name, and ID_number of each entry (3 columns/fields). The "Shape" field is an un-editable field describing the data type, which must be the same for each entry in the database.



Attributes of Level1_na.shp

Shape	Area	Name	ID_Number
Polygon	368760998528	Great Salt Lake	0
Polygon	3232000996288	Yukon	1
Polygon	1749654000192	Mackenzie	2
Polygon	3032711002496	Northwest Territ	3
Polygon	1109406998976	Saskatchewan	4
Polygon	2383419001024	Hudson Bay	5
Polygon	1055159999616	St. Lawrence	6
Polygon	1536556993408	Atlantic Coast	7
Polygon	3221209000128	Mississippi	8
Polygon	4403189008448	Mexico Coast	9

Figure 5.1 – Sample Watershed Attribute Table in ArcView

In general, tabular geographic data may be displayed entirely as in Figure 5.1, or in a summary form for each individual record. When displayed in this summary form, the content of each field is displayed for the selected record, along with the field name. These contents are the attributes of the selected record. As shown in Figure 5.2, the ID_Number attribute of the entry with the Name attribute “Mississippi” is “8”. The ID_Number attribute may contain any type of data (integer, long, string, etc), but for correct processing the field must be converted into a numerical format. In this dataset, the “Name” attribute contains the name of the main hydrologic feature of the watershed entry.

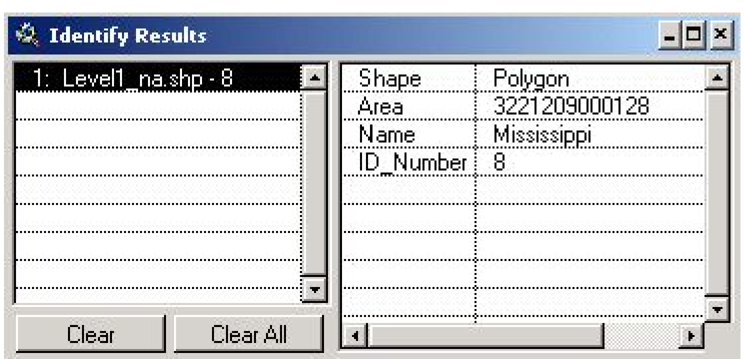


Figure 5.2 – Sample Record Summary with Individual Attributes (in ArcView)

The Pfafstetter Tools all modify the tabular attribute data of the database to which they are applied. Specifically, they create new fields in the database, and they populate these fields based on the already existing database attributes. For example, the Downstream Area Identification macro adds two new fields to a database, namely the “Downstream” field and the “Comments” field. After using this macro, each record contains a new “Downstream” and a new “Comments” attribute related to the geographic location of the database entry. This is shown in Figure 5.3, where the Downstream Area Identification macro has been applied to the database shown in Figures 5.1 and 5.2. The entry with the name “Mississippi” now has the “Downstream” attribute “-999” and the Comments attribute “Water.” These new attributes correctly signify that the watershed containing the Mississippi River drains directly to the ocean.

In order to process the information contained in the database attribute table, the Pfafstetter Tools macros all use arrays. An array is a set of storage spaces in the memory of a computer, and the spaces in this set are related based on an array **index**, or indices if the array is multi-dimensional. The information stored in each space in the array is then referred to by the index of the storage space. For example, **sample(9)** refers to the information stored in the “9” index position of the array named “sample.” It is not necessarily true that sample(9) refers to the 9^{th} entry in the “sample” array, because the first index value for an array may be set to any number desired by the programmer. In the Pfafstetter Tools, most arrays store the contents of one field in the attribute table. The first index value of each array is “1” so that the first entry in the array corresponds to a given attribute of the first entry in the watershed database. Figure 5.4 shows sample arrays that were constructed from the watersheds attribute table shown in Figure 5.3.

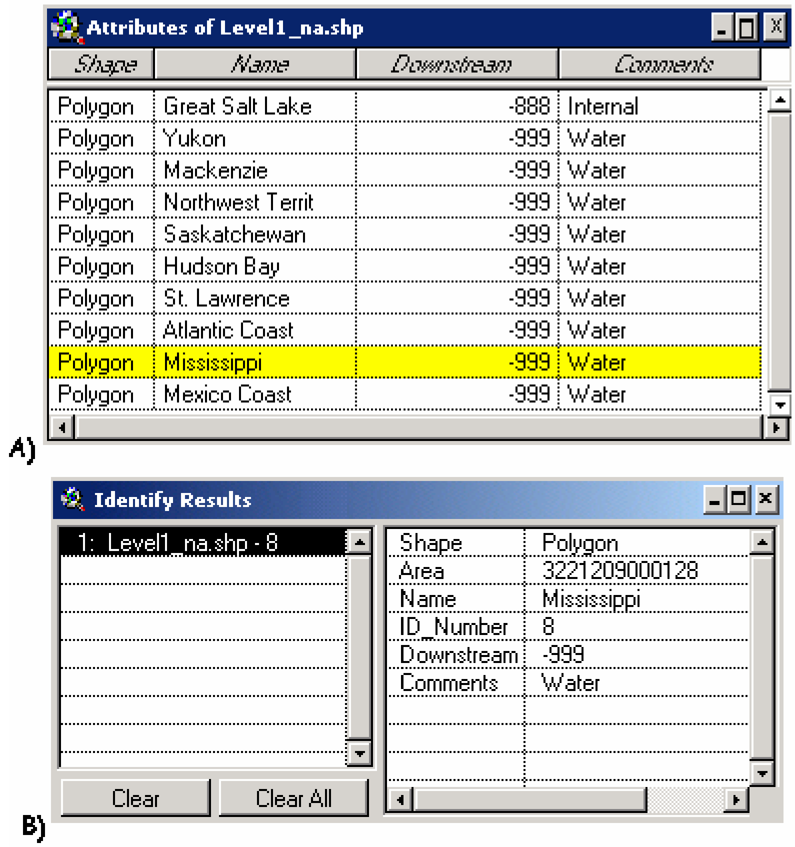


Figure 5.3 – Watershed Attributes added by the Downstream Area Identification macro – A) Attribute table showing the new "Downstream" and "Comments" fields ("Area" and "ID_Number" fields were removed for clarity), B)Record summary for the Mississippi watershed, which now contains specific "Downstream" and "Comments" attributes.

For the remainder of this work, entire arrays are first introduced in bold face type. After they have been introduced, subsequent references to the array name are made without the bold face type. If an array is referenced without any index value given in the parentheses (ex. Sample()), then the entire contents of the array are being discussed. When referring to a specific value stored in an array, the index value of the entry will be given in parentheses (ex. Sample(9)). In discussing a known array attribute with an unknown array index, the variable is referenced with a variable index (ex. Sample(I)). This variable index may take on any of the possible index values in the array in question.

Array (index)	Value
ID_number(1)	0
ID_number(2)	1
ID_number(3)	2
ID_number(4)	3
ID_number(5)	4
ID_number(6)	5
ID_number(7)	6
ID_number(8)	7
ID_number(9)	8
ID_number(10)	9

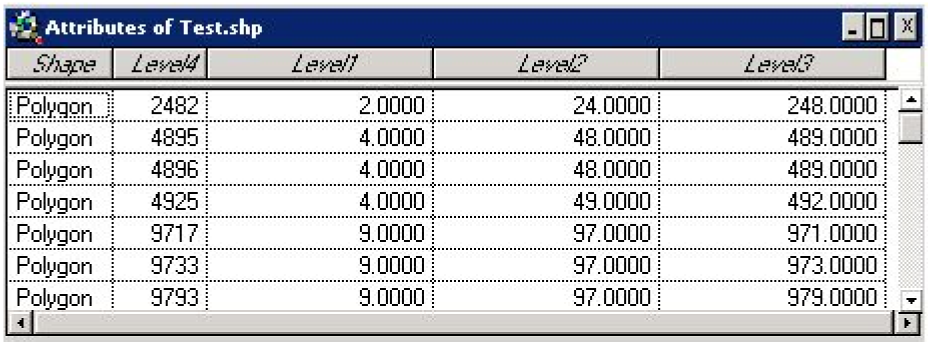
Array (index)	Value
Name(1)	"Great Salt Lake"
Name(2)	"Yukon"
Name(3)	"MacKenzie"
Name(4)	"Northwest Territ"
Name(5)	"Saskatchewan"
Name(6)	"Hudson Bay"
Name(7)	"St. Lawrence"
Name(8)	"Atlantic Coast"
Name(9)	"Mississippi"
Name(10)	"Mexico Coast"

*Figure 5.4- **ID_number()** and **Name()** Arrays from the Watersheds attribute table*

Each of the Pfafstetter Tools functions works by reading entire attribute table fields into arrays in the computer's memory. Once stored in array format, the array contents may be analyzed and manipulated to produce the desired program results. These results are written into new arrays that become new fields in the original attribute table. The individual programs employ different arrays, and the size, contents, and usage of each array is described in the function code (Appendix A).

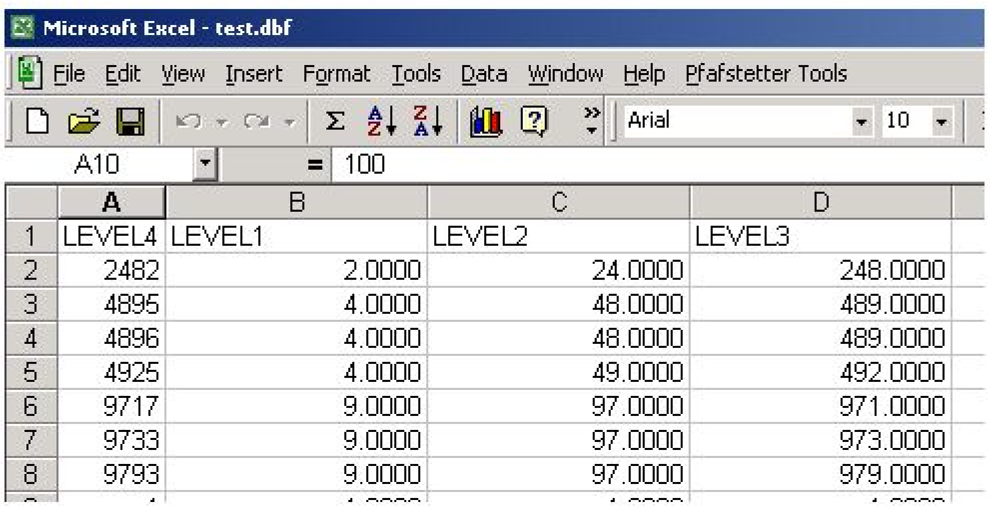
5.1.2 – Programming Concepts Specific to Microsoft Excel

Microsoft Excel is a standard spreadsheet program, which can be used to manipulate tabular data such as the attribute tables from ArcView shapefiles. Shapefiles are made up of either 3 or 5 individual files, each with the same file name but with a different extension. A sample shapefile **test.shp** contains the following files: *test.dbf*, *test.sbn*, *test.sbx*, *test.shp*, and *test.shx*. The ***.sbn*** and ***.sbx*** files are not always needed in order to display and manipulate the spatial data properly. The ***.dbf*** file is a “Database Format” file that stores the attribute data for the spatial objects contained in the other files. This file is readable by Microsoft Excel.



Shape	Level4	Level1	Level2	Level3
Polygon	2482	2.0000	24.0000	248.0000
Polygon	4895	4.0000	48.0000	489.0000
Polygon	4896	4.0000	48.0000	489.0000
Polygon	4925	4.0000	49.0000	492.0000
Polygon	9717	9.0000	97.0000	971.0000
Polygon	9733	9.0000	97.0000	973.0000
Polygon	9793	9.0000	97.0000	979.0000

A)



	A	B	C	D
1	LEVEL4	LEVEL1	LEVEL2	LEVEL3
2	2482	2.0000	24.0000	248.0000
3	4895	4.0000	48.0000	489.0000
4	4896	4.0000	48.0000	489.0000
5	4925	4.0000	49.0000	492.0000
6	9717	9.0000	97.0000	971.0000
7	9733	9.0000	97.0000	973.0000
8	9793	9.0000	97.0000	979.0000

B)

Figure 5.5 – Sample Shapefile Attribute Data – A) in ArcView, B) in Microsoft Excel

In ArcView or ArcGIS, attribute tables are shown with the field names and the field “Shape,” which contains either “Point,” “Line,” or “Polygon” for each record in the database. This field describes the type of data represented in the attribute table. When the shapefile’s **.dbf** file is opened in Microsoft Excel, the “Shape” field is not included. This is because this field is not editable in the

ArcView or ArcGIS programs. However, in Microsoft Excel, the first row in the spreadsheet contains the field names for the attribute table. These names are also not editable within ArcView or ArcGIS programs, although each may be assigned an alias by adjusting the shapefile's theme properties. Within Microsoft Excel, these field names may be changed, and the changes are recognized by ArcView or ArcGIS.

With the field names included in the Microsoft Excel spreadsheet, the actual geospatial attribute data begins in the second spreadsheet row. Therefore, in manipulating existing attribute data, only rows 2 through N are considered, where N is the number of rows in the spreadsheet. The Pfafstetter Tools macros each read columnar data from the spreadsheet into the computer memory, and the first datum is always read from row number 2. The field headings in the first spreadsheet row are used to determine which spreadsheet column contains the data desired by the macro. For example, if the macro requires Level 3 Pfafstetter codes, then the macro searches the entries in the first row until it finds an entry equal to "LEVEL3." The match is not case sensitive, although all data that is first viewed in ArcView or ArcGIS and then in Microsoft Excel will have field names in capital letters. The output from the Pfafstetter tools macros consists of fields with names given with the first letter capitalized. These field names are capitalized when the .dbf file is accessed in ArcView or ArcGIS. Once the macro has located the desired field name, then the data stored in that spreadsheet column is read into the appropriate memory array.

The number of fields in the database is determined by counting the consecutive non-null values in row #1. These values, which are the field names, may not be null, and therefore the first null value will occur in the spreadsheet column that does not contain any data from the attribute table. The number of columns is used both in locating the desired database fields within the spreadsheet and in adding new fields to the attribute table. It is also needed in order to determine the number of records within the database, which is important because it determines the size of the memory arrays required by the macros.

The memory arrays may be of any size, and their size is determined by the number of entries in the database. This is done by counting the number of non-null entries in each column in the spreadsheet and taking the largest of these numbers as the array size. In counting the number of entries in the database, it is assumed that most records will contain a non-null value for all fields. If the field contains a null value in five consecutive rows, then the number of rows is equal to the number of the last non-null record. The number of entries in the database is then this number of rows minus 1, to account for the inclusion of the field name in the spreadsheet. This 5-row factor of safety allows the macros to function properly on partially incomplete data. For example, if the database contained 1000 records, yet in the first field records #1, #2, #3, and #4 contained null values, these 4 records would be counted as long as record #5 contained a non-null value. If record #5 did contain a null value, then the macro would think that the database did not contain any records. However, it would then count the number of records in all of the other columns. If at least one of those columns is properly attributed with non-null values, then the macro would recognize that 1000 records exist in database.

Once the attribute table data is stored in the arrays within the computer memory, the macros function without any dependence upon Microsoft Excel. The macro algorithms do not make use of any of the spreadsheet functions within the Excel software. This allows the macros to be easily adapted for use with other COM compliant database management tools. After the macros have performed there calculations, new fields are added to the attribute table. This is achieved in two steps. First, the fields are written onto the Microsoft Excel spreadsheet, in the columns immediately following the right most original data column. For example, if two columns are to be added to the database, the data are added to the spreadsheet columns with number 1 and 2 greater than the original number of columns in the spreadsheet (Microsoft Excel numbers its columns from 1 to N left-to-right across the spreadsheet). The second step is to adjust the database definition in the **.dbf** file so that it includes the newly added columns. This adjustment is performed as part of the macro with the following code:

```
Range(Cells(1, 1), Cells(numrows, numcolumns + 2)).Name = "Database"
```

This code tells the computer to regard all of the cells within the specified range as part of the "Database." This range begins with the cell in the first row and column of the spreadsheet, and ends with the last entry in the newly added columns. The **numrows** variable stores the number of non-null rows in the spreadsheet, and the **numcolumns** stores the number of original fields included in the attribute table. The "2" in the code statement signifies that the macro in question added two new fields to the attribute table.

In order to view the modified attribute table in ArcView or ArcGIS, it is necessary to save the Microsoft Excel spreadsheet as a **Dbase IV (.dbf)** file with the same name as the original **.dbf** file. This file format is not compatible with many of the Microsoft Excel functions, and when attempting to save a file in this format Excel warns the user that some data will be lost. This refers to relationships between cells, and is not important for use in ArcView 3.x or ArcGIS.

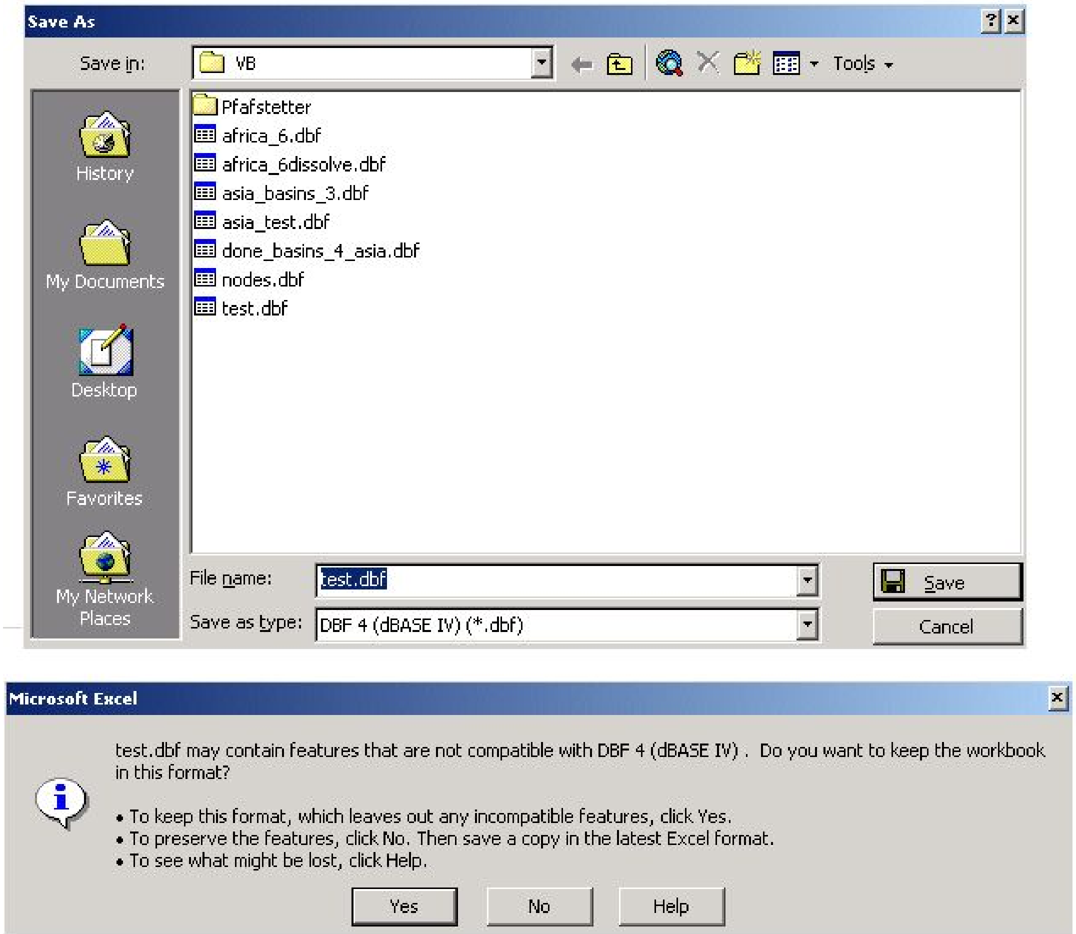
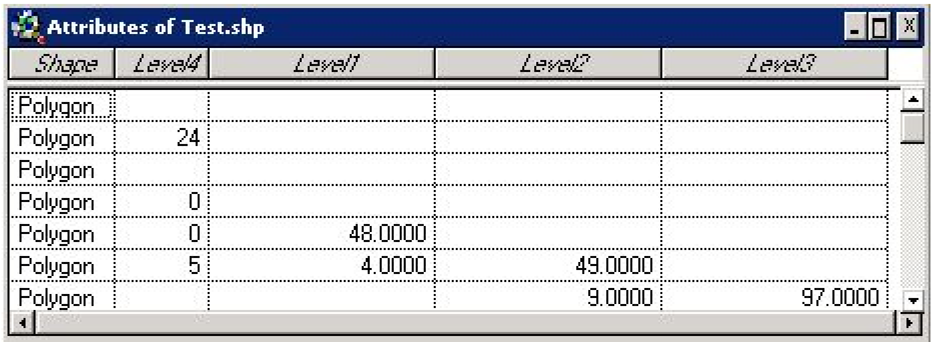


Figure 5.6 – Saving the Attributes in Microsoft Excel.

Once saved the new fields are incorporated into the shapefile attribute table, and they are viewable and accessible in ArcView or ArcGIS. However, if the shapefile is in use in ArcView or ArcGIS when the macro is applied in Microsoft Excel, then the saved and modified attribute table is not immediately viewable.

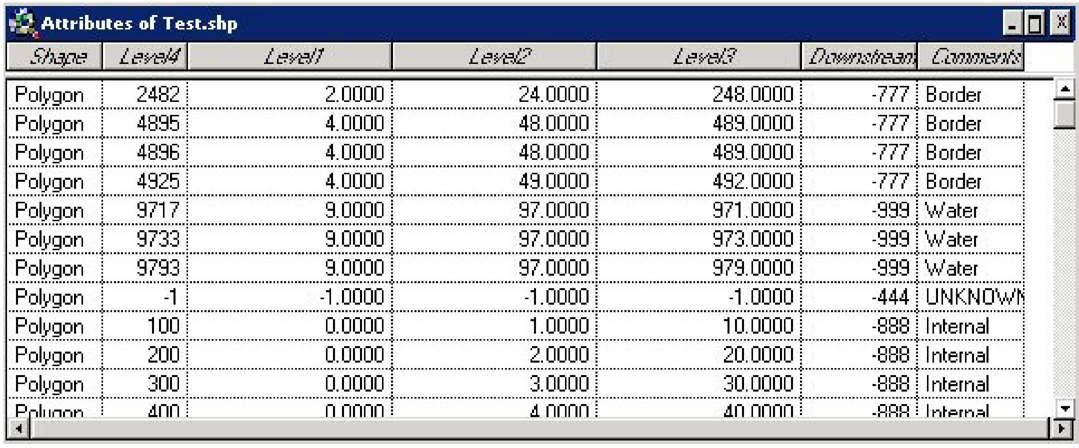

Attributes of Test.shp

Shape	Level4	Level1	Level2	Level3
Polygon				
Polygon	24			
Polygon				
Polygon	0			
Polygon	0	48.0000		
Polygon	5	4.0000	49.0000	
Polygon			9.0000	97.0000

Figure 5.7 – "Scrambled" Test.shp attribute table after it was modified in Excel

In this situation, the new attribute table will appear scrambled and incomplete (Figure 5.7). To fix this problem, it is necessary to delete the theme/layer from the ArcView project or ArcGIS map and then re-add it to the view (Figure 5.8). The new fields may be sorted and queried just as any other field in the attribute table.

The modification of the shapefile attribute table in this manner works because the order of the records in the table is not changed and records are never added. Therefore, the added field attributes will correctly correspond to the appropriate geospatial data. If the order of the records is modified, then each record will have attributes pertaining to another record in the database (Figure 5.9). This confusing situation is easily avoided if the order of the records is not disturbed within the Microsoft Excel spreadsheet.


Attributes of Test.shp

Shape	Level4	Level1	Level2	Level3	Downstream	Comments
Polygon	2482	2.0000	24.0000	248.0000	-777	Border
Polygon	4895	4.0000	48.0000	489.0000	-777	Border
Polygon	4896	4.0000	48.0000	489.0000	-777	Border
Polygon	4925	4.0000	49.0000	492.0000	-777	Border
Polygon	9717	9.0000	97.0000	971.0000	-999	Water
Polygon	9733	9.0000	97.0000	973.0000	-999	Water
Polygon	9793	9.0000	97.0000	979.0000	-999	Water
Polygon	-1	-1.0000	-1.0000	-1.0000	-444	UNKNOWN
Polygon	100	0.0000	1.0000	10.0000	-888	Internal
Polygon	200	0.0000	2.0000	20.0000	-888	Internal
Polygon	300	0.0000	3.0000	30.0000	-888	Internal
Polygon	400	0.0000	4.0000	40.0000	-888	Internal

Figure 5.8 – Attribute table including Downstream Area Identification results

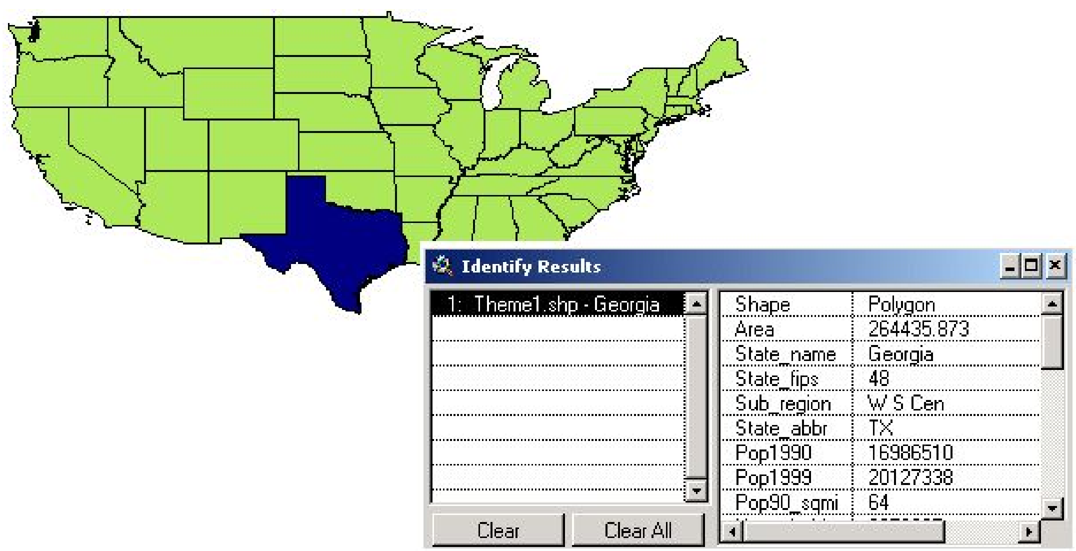


Figure 5.9 – Incorrect Attributes Due to Row Re-Arrangement in Microsoft Excel. The "State_Name" field was alphabetized in Excel without re-arranging the order of the records in the attribute table. This resulted in the Texas polygon receiving the "State_Name" attribute "Georgia." All of the other attributes are those appropriate for the Texas polygon.

5.2 The Downstream Area Identification Macro

5.2.1 Introduction

The Downstream Area Identification macro determines the area immediately downstream of each area in a dataset attributed with Pfafstetter-based numerical codes. These codes may be assigned either according to the Pfafstetter system, to the USGS-Pfafstetter system, or to the CRWR-Pfafstetter system discussed in Chapters 6 and 7. For this discussion, the phrase “Pfafstetter-based” signifies that numerical IDs were assigned to drainage areas based on one of these system methodologies.

The macro assigns to each area in a database the Pfafstetter-based code of the area immediately downstream. The assignment is made without reference to any spatial information within a GIS system, and is only dependent upon the topology implied by the numerical Pfafstetter-based codes. The assigned downstream code is stored as the “Downstream” attribute of each area in the database. The macro is really a series of connected subroutines (Figure 5.10), each of which determines the downstream ID for a given type of area. The subroutines are applied sequentially to each area in the database until the downstream characteristics of that area are determined. Once each determination is made, the macro then determines the downstream characteristics of the next entry in the database. This continues until all of the database entries have had their downstream characteristics determined.

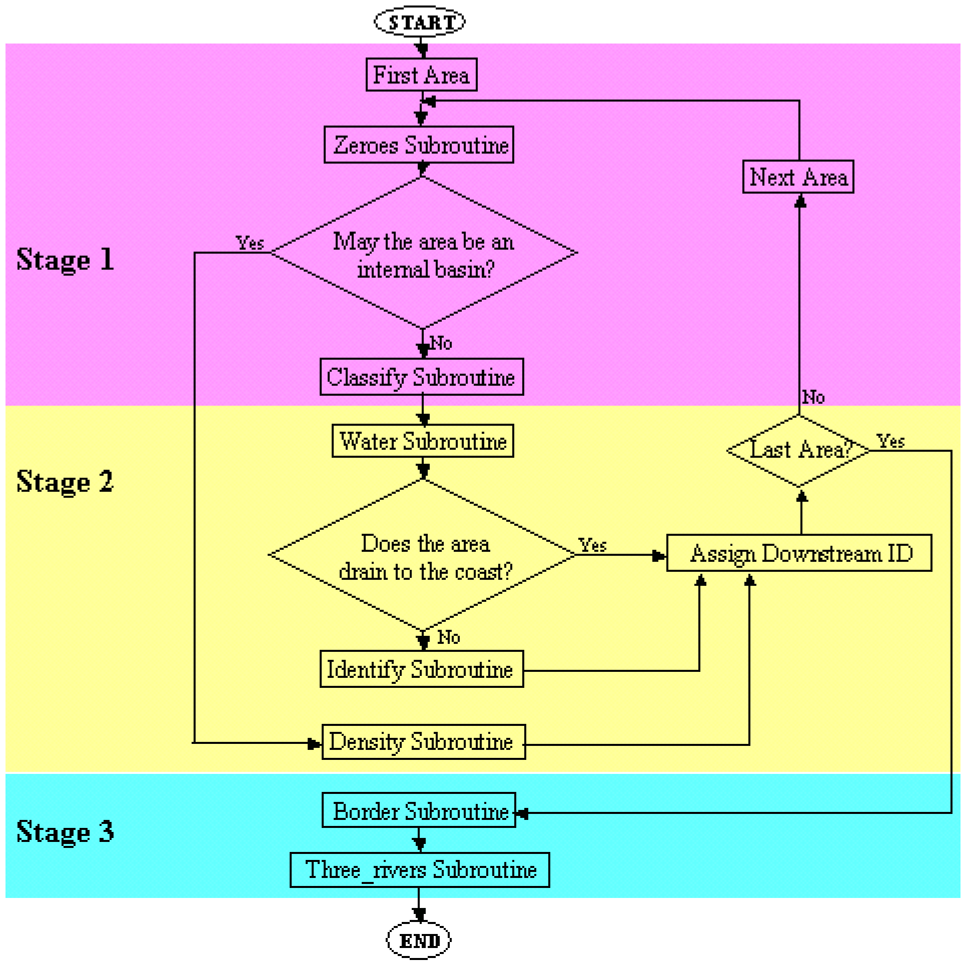


Figure 5.10 – Downstream Area Identification Flow Chart

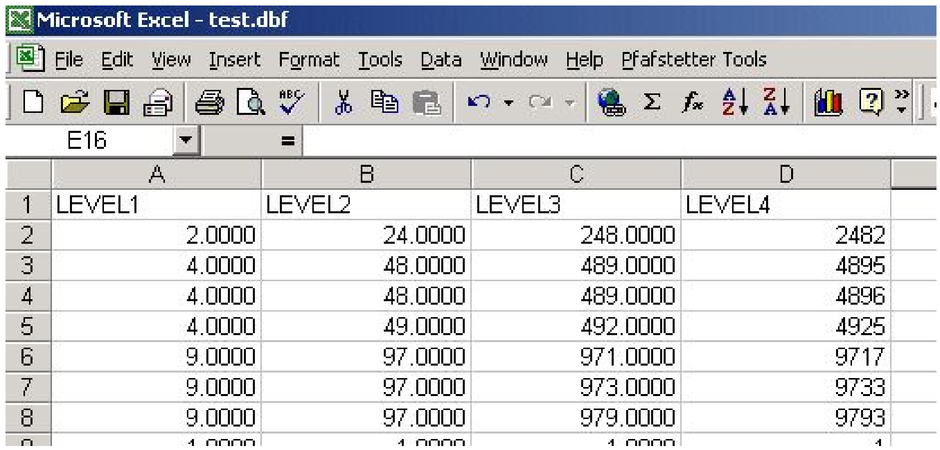
After obtaining the input data from the spreadsheet, the macro first separates each area's Pfafstetter code into its constituent digits. Once this data pre-processing is complete, the Downstream Area Identification process begins with the first area in the database. In the first stage of the process, the **zeroes** and **classify** subroutines prepare each drainage area's code for further analysis. The zeroes subroutine determines if the area may be an internal basin, and the classify subroutine categorizes the digits of the Pfafstetter-based code so that the downstream determination may be made. The second stage of macro operation includes the **water**, **identify**, and **density** subroutines, which determine if an area drains directly to the coast, if an area drains to another areas in the database, or if the area is an internal basin. If the area drains to another area in the database, then this area is identified. Finally, in the macro's third stage the **borders** and **three_rivers** subroutines check the downstream determinations for any errors and inconsistencies. This last step is important especially if the function is applied to a subset of a Pfafstetter attributed database. Each of these subroutines is discussed in detail in the following subsections.

The Downstream Area Identification macro will correctly identify the downstream elements of areas classified up to Pfafstetter level 15. This limit is not a defect of the macro itself, but rather a limit of Microsoft Excel to store and process integers with more than 15 digits. This integer-digit limitation is unlikely to provide significant hardship for users, for the current HYDRO1K dataset only contains up to the Level 6 watersheds – a full 9 levels below the current maximum capabilities of Excel. However, refined datasets, such as the EDNA dataset for the United States, may be so detailed that they approach the limits of the function. One possible method for increasing the capacity of the Downstream Area Identifier macro would be to split multi-digit Pfafstetter codes into a set of smaller code sections. For example, a level 16 Pfafstetter code could be split into two 8-digit code sections. The Downstream Area Identifier macro would have to be modified slightly to account for this scenario, but it would not be a difficult modification to make.

5.2.2 Data Requirements and Pre-Processing

In order to carryout the Downstream Area Identification macro, the shapefile attribute table must contain a field storing Pfafstetter codes, and each code in the field must pertain to the same Pfafstetter level. The header for this field must be of the form "LEVELX" where "X" is a positive integer corresponding to the Pfafstetter code level. Capitalization of the letters in "LEVEL" is not necessary.

The attribute table may contain other fields, and it may contain the Pfafstetter codes for multiple levels.



	A	B	C	D
1	LEVEL1	LEVEL2	LEVEL3	LEVEL4
2	2.0000	24.0000	248.0000	2482
3	4.0000	48.0000	489.0000	4895
4	4.0000	48.0000	489.0000	4896
5	4.0000	49.0000	492.0000	4925
6	9.0000	97.0000	971.0000	9717
7	9.0000	97.0000	973.0000	9733
8	9.0000	97.0000	979.0000	9793

Figure 5.11 – Sample shapefile attribute table for Downstream Area Identification. The table must contain at least one field of Pfafstetter codes (LEVELX) where "X" is the level number. (Note: Due to clarity and space concerns, only LEVEL4 data will be displayed for the remaining discussion)

Before determinations are made, Pfafstetter codes for the desired level are stored in a one dimensional array, the **totalcode()** array. The user is prompted to select the desired level for the determination in a standard Visual Basic input box.

Figure 5.12 – Pfafstetter level selection window

The provided input is stored as the **basinlevel** variable, and must be an integer greater than zero. If the selected level is found in the attribute table, the codes are written to the totalcode() array and the processing continues. If the selected level is not found in the dataset, the user is given the opportunity to select another level or to quit the program.

Once the Pfafstetter codes are stored in the totalcode() array, the processing begins by breaking each code into its constituent digits. Each digit corresponds to the area location within the lower level area. For example, for the totalcode element "1234," the "1" is the Pfafstetter level 1 digit, the "2" is the Pfafstetter level 2 digit, etc. The level two digit describes the area's position within the area uniquely identified by the level 1 digit "1." Incidentally, the name "totalcode" was given to the array because the array stores the entire Pfafstetter code of an area as one number. Subsequently, the **levels()** array stores the digits of each totalcode as distinct units, one unit for each level up to the selected basin level. The **levels()** array is a two dimensional array, where the first dimension is the index variable and the second dimension is the Pfafstetter level number.

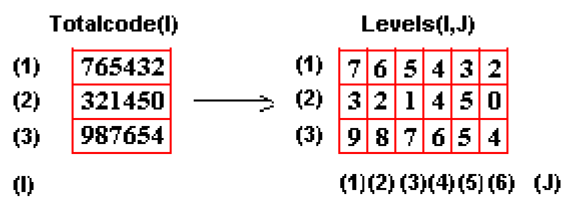


Figure 5.13 – Conceptual representation of totalcode() and levels() arrays

The size of the levels() array is determined both by the number of areas in the database and by the level selected for analysis (stored as the basinlevel variable). In Figure 5.13, the analysis level is level 6, and therefore the levels() array stores values for J = 1 to 6 for each totalcode() element I. The elements within the levels() array are determined from the totalcode() array with the **digits** and **rounddown** subroutines. These subroutines make use of basic multiplicative principles of integer and floating point numbers, further description of these subroutines is not warranted in this work. However, it should be noted that these functions are not hampered by the 15-digit limitation of Microsoft Excel. The program codes for these subroutines are given in Appendix A.

After the codes are broken down into digits and stored in the levels() array, the downstream determination begins. The Downstream Area Identification macro uses the information from the totalcode() and levels() array to determine the downstream area codes, which it stores in the **Dwnstrm()** array. All elements in the Dwnstrm() array initially contain the value –11111. This value is an arbitrarily chosen number that will not equal the Pfafstetter code of an area in the database. Along with the Dwnstrm() array, the function also produces information related to the classification of each area into various drainage categories. These categories are:

- "Internal" Drainages – areas that do not drain, i.e. internal sinks
- "Water" Drainages – areas that drain directly to the ocean
- "Border" Drainages – areas that drain to the border of the study area, and
- "Success" Drainages – areas that drain to other areas in the database

This categorical information is stored in the **Comments()** array. The Comments() array is also useful in assessing the accuracy of the Downstream Area Identification process. Other possible entries in the Comments() array are:

- Check – Internal
- Check – Border, and
- Unknown

These entries identify their respective areas as areas for which the downstream element may not be identified with 100% certainty. This only occurs if the Pfafstetter codes were assigned incorrectly, or occasionally if the function is run on a subset of Pfafstetter attributed data. (Application of the Pfafstetter Tools to subsets is discussed in Section 5.2.5). If the Pfafstetter codes are correctly assigned and if the function is run on the entire database, then all downstream elements are determined with 100% certainty. Both the Dwnstrm() array and the Comments()

array contain one value for each area in the database. The Downstream Area Identification procedures are different for each of the aforementioned drainage area categories, and each procedure is described in a separate sub-section.

The first step in the pre-processing for the Downstream Area Identification process to determine if the dataset contains any unrecognizable codes. Some examples of invalid codes are negative numbers, floating point numbers, and numbers containing more digits than the analysis level selected by the user. If an area has such a code, the pre-processing algorithms assign the value "-444" as the Dwnstrm(I) attribute, and "UNKNOWN" for the Comments(I) attribute. Upon program completion, the user can check the database in order to determine why any area codes were unrecognizable, and the codes can be corrected if necessary. In the HYDRO1K dataset, only the miscellaneous 1-cell areas are given negative values. These areas are merely remnants of the vectorization process of the raster-defined drainage areas, and are inconsequential in the overall database. They also do not have defined drainages or downstream areas. They are often located in low-lying coastal areas where it is extremely difficult to define drainage boundaries from coarse raster data.

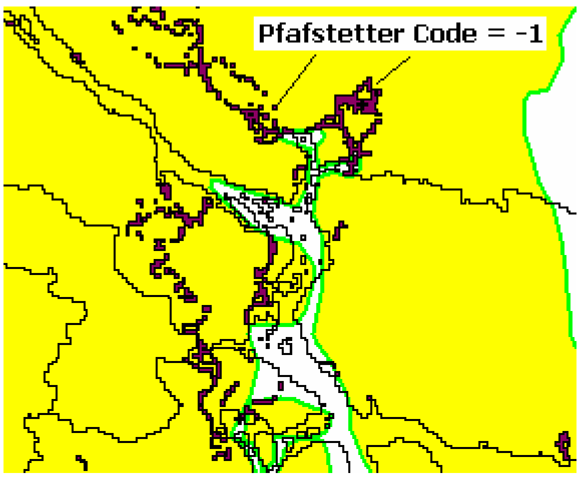


Figure 5.14 – Areas with Negative Pfafstetter Codes (Purple) – Located along low-lying coastal areas. The areas shown are from the HYDRO1K data for Bangladesh.

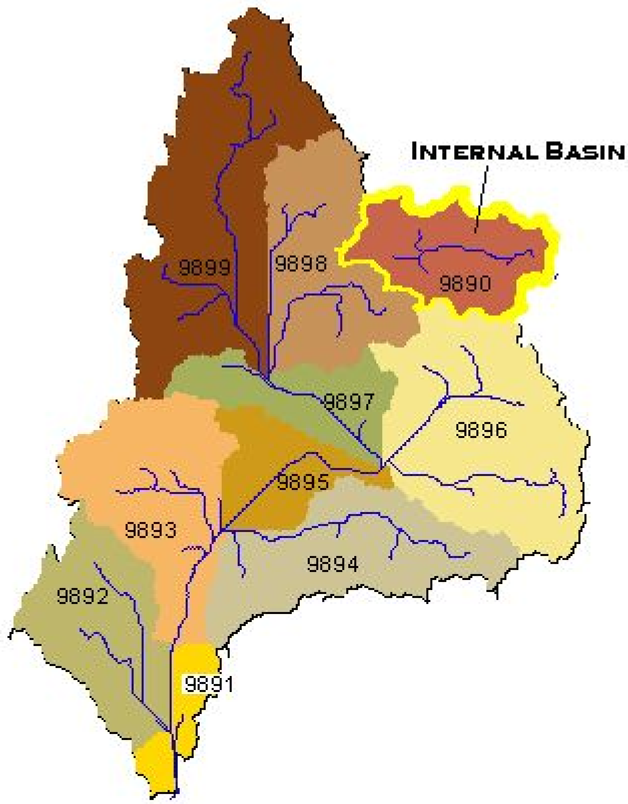


Figure 5.15 – A properly coded internal basin in HYDRO1K data – area 9890 (Yellow outline) does not drain into another area and it has a "0" final digit.

The final step in preparation for determining the downstream areas is to distinguish those areas whose code contains the digit "0." According to the Pfafstetter system theory, a "0" highest-level digit signifies that an area is an internal basin. Such basins will not have any downstream basins. For example, internal basin 9890 in the HYDRO1K dataset for North America does not drain to any other area, and it is appropriately numbered with a zero at it's level 4 digit (Figure 5.15).

An apparent contradiction to this theory is evident in the codes assigned to HYDRO1K drainage areas. In the USGS-HYDRO1K numbering system, areas are assigned the value "0" at the highest level digit if the areas lack sufficient drainage density to support further subdivision into higher Pfafstetter levels. For example, at the level 3 scale, interbasin 941 in the Asia HYDRO1K dataset is properly numbered (Figure 5.16). Yet, at the level 4 scale, the area is labeled 9410 because there are not any tributaries available to divide the level 3 interbasin into nine or ten smaller level 4 areas. For proper Pfafstetter coding assignments, four tributaries are needed for each area subdivision into higher-level areas. In HYDRO1K data, if four tributaries are not available, then sub-divisions are not created and the area is given the "0" digit at the end of its lower level code.

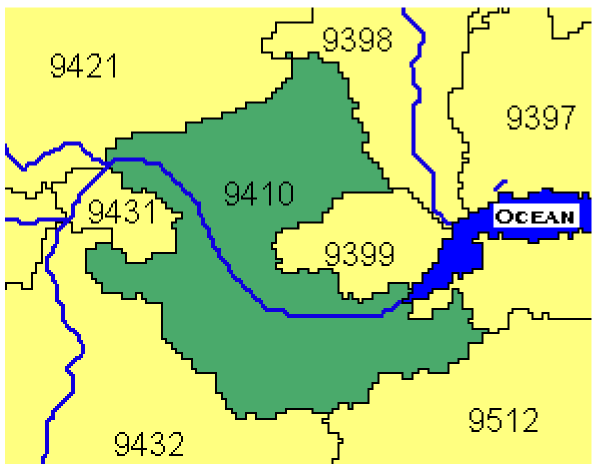


Figure 5.16 – Insufficient "Drainage Density"- Area #941 does not contain enough tributaries to support proper level 4 subdivisions. Area 9410 has a "0" fourth level digit when it is obviously an interbasin (HYDRO1K).

A second complication arises in that internal basins may be further subdivided into sub-areas at a next higher Pfafstetter level. In such instances, by convention the newly divided basins would be numbered in a clockwise fashion starting from the most northern basin. The numbering progresses so that basin XY02 would border XY01 and XY03, as occurs when assigning level 1 codes. In such a situation, area XY02 would not drain to XY01, as would happen if the level 3 digit were not "0." This means that in determining the downstream basin, it is insufficient just to focus on the final digit if the Pfafstetter code contains one or more “0” digits. Having a “0” digit in the Pfafstetter code disrupts the automatic downstream determination process by providing exceptions to the general coding patterns involved in the implied topological relationships between areas. Because of this fact and the HYDRO1K use of “0” digits to signify drainage density problems, USGS-Pfafstetter codes containing “0”’s at any digit warrant special consideration. This special consideration is carried out in the first and second stages of the Downstream Area Identification process.

5.2.3 Stage #1 – The Zeroes and Classify Subroutines

Within stage #1, each area’s code is analyzed in order to determine which subroutine will identify the area’s downstream area. The first subroutine, **zeroes**, determines whether an area may be an internal basin. If the area may not be an internal basin, then the **classify** subroutine is run. This subroutine categorizes each digit in the area’s code in order to hasten the downstream area identification process.

The zeros subroutine is an uncomplicated procedure that makes use of the levels() array. For each area in the dataset, it checks each of the digits in the levels() array. If at least one of these digits is “0,” the Dwnstrm(I) value for that area is changed from its initial value of –11111 to –555. This value, -555, is not possible in the Pfafstetter system and was chosen arbitrarily. The subroutines that carry out the Downstream Area Identification process for areas with codes lacking “0” digits will run only if the Dwnstrm(I) attribute is equal to -11111. Areas containing “0” digits have their downstream areas identified with the **density** subroutine. The density subroutine only runs on areas with the Dwnstrm(I) attribute equal to –555.

If an area’s code does not contain a “0” digit, then the code is analyzed within the **classify** subroutine. Downstream determinations are made based on an analysis of the digits in a given each area’s Pfafstetter code. This analysis starts with the **classify** subroutine, which is called every once for each area in the database. The classify subroutine looks at each digit of the area's code (as stored in the levels(I,J) array) and classifies the digits as pertaining to one of four categories. The results of the classification are stored in the **test()** array, which is a one dimensional array containing **basinlevel** elements. For each digit, the test(I) value may be either:

- Odd not equal to 1 – Denoted as "3"
- Even not equal to zero – Denoted as "2"
- Zero – Denoted as "0", or
- One – Denoted as "1".

These four categories are important because they dictate the possible relationships between adjacent downstream and upstream areas. Digits within each category imply the same relationship between drainage areas, whereas the relationships implied by each category are different. For example, the digits "2" and "4" both imply that the area is a basin draining to an interbasin. This implied relationship is different than that suggested by the digits "3" and "5." These digits both imply that the area is an interbasin draining to another interbasin. In the Pfafstetter system, the "0" digit implies the area is an internal basin, and the "1" digit implies the area is the most downstream area within the lower level area. Determining the downstream area for areas with codes containing these digits requires a more detailed analysis than is required for the other two categories. With this classification scheme, it is easy to see that different Pfafstetter-based codes may imply identical drainage relationships (Figure 5.17).

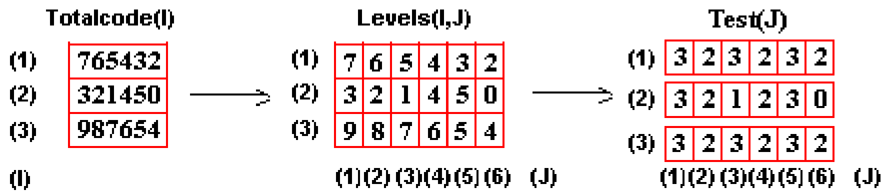


Figure 5.17 – Creating the Test() array – note that the relationships implied by Totalcode(1) and Totalcode(3) are identical

5.2.4 Stage #2 – The Water, Identify, and Density Subroutines

Stage #2 of the Downstream Area Identification process is the stage in which most downstream determinations are made. The stage may be divided into two separate parts, and each area has its downstream area determined by one of the two parts. For areas with codes containing “0” digits (as identified by the zeroes subroutine in Stage #1), the downstream characteristics are determined with the **density** subroutine. This subroutine makes up one part of the Stage #2 processes. For areas with codes without “0” digits, the classify subroutine in Stage #1 prepares the codes for analysis in the **water** and **identify** subroutines of Stage #2. These two subroutines make up the other part of the Stage #2 processes.

After the classify subroutine, the second stage of the Downstream Area Identification process begins with the **water** subroutine. This subroutine determines if an area is a coastal area by testing the area’s code against the three criteria for coastal areas described in Section 4.3.4. If the code is found to meet one of the criteria, then the area is assigned the Dwnstrm(I) attribute “-999” and the Comments(I) attribute “Water.” Once these assignments are made, then the entire Downstream Area Identification process starts over with the next area in the database. However, if an area does not meet any of the three coastal area criteria, then its downstream area is determined by the **identify** subroutine.

The **identify** subroutine determines the downstream area for non-coastal areas. These areas have codes that do not do not contain zeros, and that do not meet any of the ocean-draining criteria. The water subroutine must be run before the identify subroutine because ocean draining areas are not distinguished from other areas in the identify subroutine's downstream determination process. If the identify subroutine is run before the water subroutine, than the ocean-draining areas have their downstream areas determined incorrectly.

The identify subroutine determines downstream areas based on a numerical analysis of the Pfafstetter codes. This analysis follows directly from the methodology presented in Section 4.3.3 for non-coastal areas in the Pfafstetter system. Specifically, if an area is an interbasin with its highest-level digit not equal to "1, " then the identify subroutine determines the downstream area to be the area with the code of the interbasin minus two. If the area is a basin (with an even highest-level digit) the downstream area has a code equal to the code of the basin minus one. This follows directly from the detraction equations (eqn. 4.1-4.3). However, because the USGS-Pfafstetter system assigns "0" digits to drainage deficient areas, and because these areas may be downstream of non-coastal areas, use of the detraction equations is no longer always valid for areas with highest-level digits equal to "1."

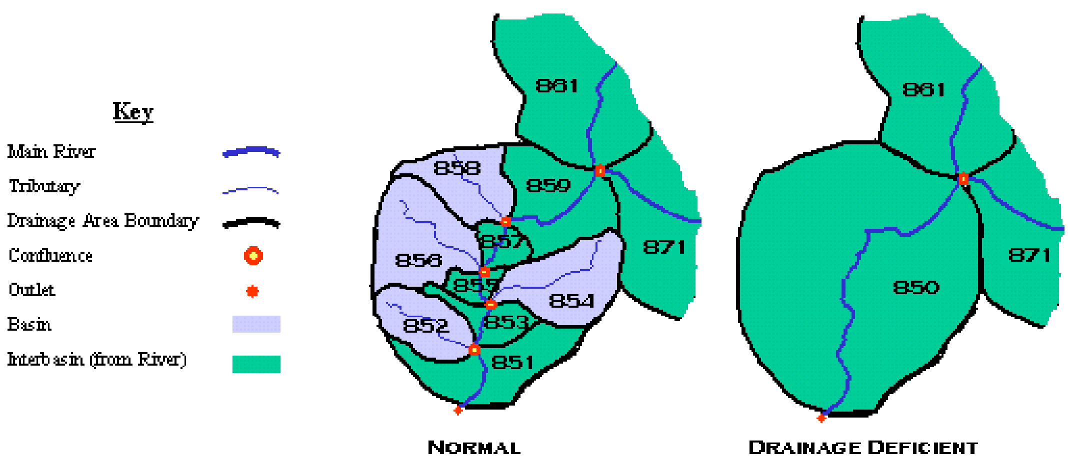


Figure 5.18 – Downstream Areas in the USGS-Pfafstetter System – The detraction equation does not apply for areas 861 because of the possibility that the downstream area is drainage deficient.

The principle behind the detraction equations is that the downstream area must be determinable mathematically. This is not the case within the USGS-Pfafstetter system because there now exist two possibilities for the code of the downstream area. The first possibility is the code predicted from the detraction equations. The second possibility is the code of a drainage deficient area. For example, the detraction equations predict that the area downstream of area #861 is area #859. However, if the level 2 area #85 is drainage deficient, then at the level 3 scale in the USGS-Pfafstetter system this area becomes area #850 (Figure 5.18). In this case, area #859 would not exist in the database, and the detraction equations would not accurately determine the correct downstream area.

In order to assigned downstream IDs to non-coastal areas with “1” highest-level digits, the identify subroutine must determine whether the lower level downstream area is drainage deficient. This determination is made by considering a range of possible downstream area numbers, and then selecting the largest number out of this range. For example, with interbasin 861, the subroutine considers all drainage areas with numbers greater than 849 but less than 860. If the interbasin 859 exists in the database, it is selected. If interbasin 859 does not exist, the next highest interbasin in the stated range is selected. As implemented in the USGS-Pfafstetter system, if interbasin 859 does not exist, it is because the level 2 area #85 is drainage deficient. Therefore, at the level 3 scale area #85 becomes area #850, and this is only area in the database with a code within the stated range. This area is identified as the area downstream of interbasin 861. In the section “Suggested Alterations to Pfafstetter System” (Section 6.5), the utility of the making downstream determinations based on search ranges is demonstrated further.

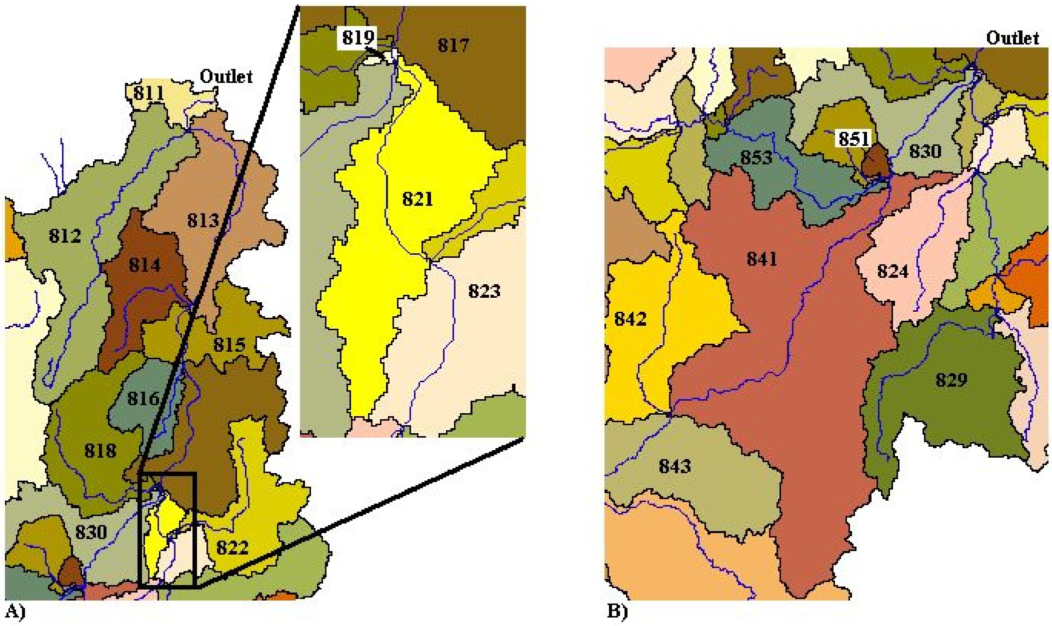


Figure 5.19 – Downstream areas determined from ranges – A) Interbasin 819 is downstream of interbasin 821 (Range: 809 < X < 820), B) Area 830 is downstream of interbasin 841 (Range 829 < X < 840). HYDRO1K data for Asia

The determination of the range in which to search for downstream area is made by considering the lower level digits of the Pfafstetter code. The first step is to consider the second highest Pfafstetter digit. If this digit is even , the acceptable downstream areas will have the codes in the range:

$$\text{totalcode}(I) - 1 > \text{Dwnstrm}(I) > \text{totalcode}(I) - 12$$

For area #861, the appropriate ranges is thus:

$$861 - 1 > \text{Dwnstrm}(I) > 861 - 12$$

$$860 > \text{Dwnstrm}(I) > 849$$

Therefore, the ten values that fall within this range are:

$$\text{Dwnstrm}(I) = \{850, 851, 852, 853, 854, 855, 856, 857, 858, 859\}$$

At the next lower Pfafstetter level, the drainage area is part of a basin, which drains to an interbasin with the code 1 less than its own. Using the example of level 3 area #861, at the level 2 scale, area #86 will drain into area #85 (See Category A, Section 4.3.3). Area #85 encompasses the level 3 areas with codes between #850 and #859. One of these areas must be downstream of area #861, so the range of possible downstream codes must include all of the numbers between and excluding 849 and 860. These bounds on the range are obtained by subtracting 12 and subtracting 1, respectively, from the code of the area in question, namely #861.

Once the acceptable range is identified, each entry in the database is tested to see if it falls into the range. For each entry that is in the range, the entry is compared with the largest entry previously found in the range. The larger value is stored and tested against other values found to be in the range. Once all database entries are considered, the largest value identified in the range is recognized as the downstream area code.

The range determination procedure is similar if the 2nd highest level digit is odd and not equal to 1, however the range of interest is shifted downward by ten:

$$\text{totalcode(I)} - 11 > \text{Dwnstrm(I)} > \text{totalcode(I)} - 22$$

For area #851, the appropriate ranges is thus:

$$871 - 11 > \text{Dwnstrm(I)} > 871 - 22$$

$$860 > \text{Dwnstrm(I)} > 849$$

Therefore, the ten values which fit this range are:

$$\text{Dwnstrm(I)} = \{850, 851, 852, 853, 854, 855, 856, 857, 858, 859\}$$

This is the same range containing the downstream element of area #861. This stems from the fact that at the level 2 scale, areas #86 and #87 both drain to area #85.

The final scenario is that the 2nd highest level digit is equal to 1. For this situation, the 3rd highest level digit must be considered. If this digit is also 1, then the subroutine considers the 4th highest level digit. Successively lower level digits are examined until a non-“1” digit is found. Eventually the program will encounter a digit not equal to 1, for if all digits were 1, the interbasin would drain to the ocean. The downstream area for such an area would have been determined with the water subroutine, and the interbasin in question would not have reached the identify subroutine in the program code. The method for making this determination is identical to the methods already discussed with the exception that the target search range is now larger and is based on the number of digits equal to “1” at the end of the Pfafstetter code in question.

The range for this situation is determined by counting the number of consecutive “1” digits starting from the highest level digit in the Pfafstetter code. It is also dependant upon whether the first non-“1” digit is “even” or “odd not equal to 1.” If it is “odd not equal to 1,” then the upper bound of the search range is given by the function:

$$\text{Upper Bound} = \text{Totalcode(I)} - \text{Detraction}$$
$$\text{Upper Bound} = \text{Totalcode(I)} - \sum_{\mathrm{i}=0}^{\mathrm{K}} 10^{i} \qquad \text{Eqn. 5.1}$$

where K is the number of consecutive “1” digits at the end of the Pfafstetter code. This equation for the upper bound of the search range is nearly identical to the one of the detraction equations presented in Section 4.3.3. The difference is that eqn. 5.1 produces a detraction term one less than that produced from equations 4.1-4.3. This one number difference is accounted for in the “less than” term in the stated search range. The lower bound of the search range is given by the function:

$$\text{Lower Bound} = \text{Totalcode(I)} - \left[2 \cdot \text{Detraction} - \sum_{i=1}^{K} 10^{i} \right]$$
$$\text{Lower Bound} = \text{Totalcode(I)} - \left[2 \cdot \sum_{\mathrm{i}=0}^{\mathrm{K}} 10^{i} - \sum_{i=1}^{K} 10^{i} \right] \qquad \text{Eqn. 5.2}$$

This lower bound is determined with the “Detraction” term calculated when determining the upper bound. The repeated use of the “Detraction” term simplifies the lower bound determination process. For example, if the interbasin in question has the level 7 Pfafstetter code 9863111, then K = 3 and the upper bound of the range equals:

$$9863111-10^{3}-10^{2}-10^{1}-10^{0} = 9862000$$

The “Detraction” term is equal to “1111.” The lower bound is therefore:

$$986311\text{-}\{2^{*}\ \text{Detraction - }(10^{3}+10^{2}+10^{1})\}$$

$$9863111\text{-}\{2 * (1111)\text{-}(10^{3}+10^{2}+10^{1})\}$$

$$9863111\text{-}\{2222\text{-}1110\} = 986099$$

To see why this is the appropriate range, consider the level 4 digit of the #986311 Pfafstetter code. The “3.” signifies that at the level 4 scale, the level 7 area was part of interbasin 9863, and it must have drained to the “1” interbasin, namely 9861. This “1” interbasin underwent 3 subdivisions to reach the 7th Pfafstetter level, therefore potentially 10^3 new areas were recognized. The search range must encompass this entire 1000 number set, starting from 9861000 and going to 9861999, inclusively.

The search range for an interbasin with an even highest level “non-1” digit is given by the following equations:

$$\begin{aligned}&\text{Upper Bound} = \text{Totalcode(I) -} \sum_{\text{i=0}}^{\text{K-1}} 10^{i}\\&\text{Lower Bound} = \text{Totalcode(I) -} \left[\left(10 \cdot \sum_{\text{i=0}}^{\text{K}} 10^{i}\right) + 2\right]\end{aligned} \qquad \text{Eqn. 5.3}$$

where K is the number of consecutive “1” digits, starting from the highest level digit. For example, if the interbasin in question had the Pfafstetter code 9862111, then K = 3 and the upper bound of the range would equal:

$$9862111\text{-}10^{2}\text{-}10^{1}\text{-}10^{0} = 9862000$$

The lower bound becomes:

$$9862111\text{-}\{10^{*}[10^{2}+10^{1}+10^{0}]+2\} = 9860999$$

This is the same range containing the downstream area of area #9863111. This stems from the fact that at the level 4 scale, areas #9863 and #9862 both drain to area #98611. Once the range is determined, the determination of the downstream area is made with the procedure already described.

If the identify subroutine determines the downstream area for an entry in the database, then the entire Downstream Area Determination process starts over with the next entry in the database. This next entry begins at Stage #1 with the **zeroes** subroutine. If the entry has a code containing a “0” digit, then the entry may be an internal basin. Its downstream characteristics are determined with the **density** subroutine, which is a separate part of the Stage #2 Downstream Area Identification processes.

The **density** subroutine determines the downstream characteristics of areas with codes containing zeros. This subroutine only considers those areas with codes containing zeros, and is therefore not applied to all areas within the dataset. The subroutine first considers those areas containing a “0” digit at the highest Pfafstetter level. Within the USGS-Pfafstetter system, such an area may be an internal basin, or it may be a drainage deficient area. This distinction is necessary only for existing HYDRO1K data attributed according to the USGS-Pfafstetter system, because only in this system are “0” digits assigned to areas with insufficient drainage densities to support higher level area divisions. Areas with insufficient drainage densities are more common at the higher Pfafstetter levels.

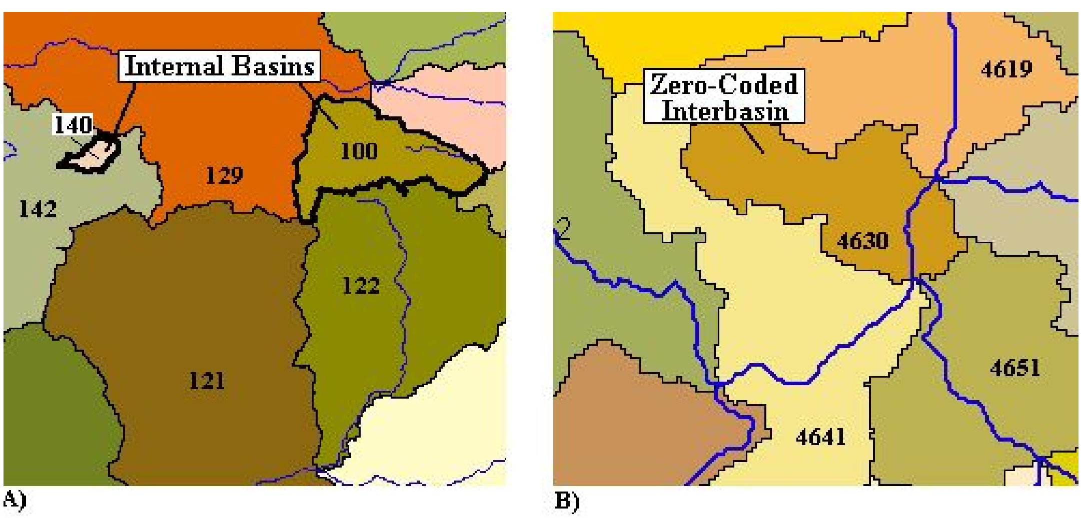


Figure 5.20 – Areas with zero-containing codes A) internal basins, B) a drainage deficient interbasin (HYDRO1K Data for Asia)

According to the Pfafstetter system theory, only internal basins may have "0" values in the highest level Pfafstetter digits. Clearly there is a conflict with this theory and the system implementation in HYDRO1K, as shown in Figure 5.20 with interbasin 4630. Despite this apparent inconsistency in methodology and theory, it is still possible to distinguish internal basins from "drainage-deficient" areas through careful inspection of the other areas in the database.

An internal basin, according to pure Pfafstetter theory, exists as a 10th subdivision of a lower level drainage area (or the 10th area in Level1). As shown with area #140 in Figure 5.20a, such an area will be geographically located near areas with Pfafstetter codes equal to itself, plus a number from 1 to 9 (Only areas #142 is shown in the figure). These areas, as a whole, make up a common lower level drainage area. A "drainage-deficient" area, as in Figure 5.20b, makes up the entire lower level drainage area, and the database does not contain areas with codes equal to the deficient area's code plus 1-9. Therefore, in determining whether a given area is drainage-deficient or internal based upon the area's code, it is necessary to determine if there exist in the database any areas with codes equal to the given code plus 1–9. The existence of any one of these areas requires that the given area be an internal basin. As such, the given area will not have a downstream area, and the Dwnstrm(I) field is attributed with the code "-888," and the Comments(I) field with "Internal."

However, if the database does not contain areas with codes equal to the given code plus 1-9, the area in question must be a drainage-deficient area. For such an area, the downstream area must be determined using methods similar to those in the **identify** and **water** subroutines. These subroutines have been modified, and the modified versions are included within the density subroutine. As with the un-modified versions, the "modified-water" algorithms must be run before the “modified-identify” algorithms in order to yield correct downstream area determinations.

The “modified water” algorithms compare the code of the area in question against the code criteria for an area draining to the ocean (Section 4.3.4). However, the “0” digit (or digits) at the end of the code are ignored. For example, the “modified water” algorithms would treat the code “9341000” as “9341,” and they would recognize this area drains to the ocean because it satisfies the ocean-draining criterion #3. An area with the code “832400” would be considered as “8324,” which does not meet any of the ocean-draining criteria. The downstream area for this area is then determined by the “modified-identify” algorithms.

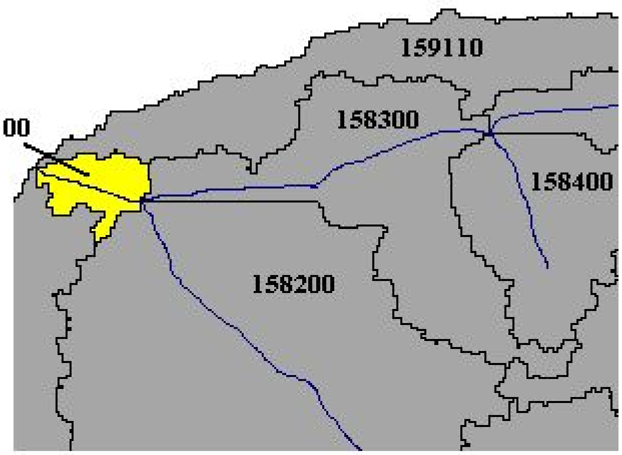


Figure 5.21 – “Modified” Water and Identify Algorithms within the Density Subroutine. Area 158100 drains to the ocean because “1581” meets criterion #3 for ocean-draining areas. Area 158300 drains to area 158100 because this code is the highest number in the dataset out of the range of numbers that could be downstream of area 158300. (HYDRO1K Data for Africa)

The “modified-identify” algorithms function differently than the “modified-water” algorithms in that they do not ignore the higher level “0” digits. Instead, these digits are counted and used to determine the numerical search range within which the downstream area code must exist. As in the identify subroutine, the search range is dependent upon the even/odd characteristic of the highest level non

“0” digit in the Pfafstetter code. With K equal to the number of consecutive “0” digits starting from the highest-level digit and the highest non “0” digit being odd not equal to 1, the upper and lower bounds of the search range are given as:

$$\text{Upper Bound} = \text{Totalcode(I)} - 10^{K}$$

$$\text{Lower Bound} = \text{Totalcode(I)} - (2 * 10^{K}) - 1$$

For area 158300 (as in Figure 5.19), K = 2 and the search range is defined as:

$$\text{Totalcode(I)} - (2 * 10^{K}) - 1 < \text{Dwnstrm(I)} < \text{Totalcode(I)} - 10^{K}$$

$$158300 - (2 * 10^{2}) - 1 < \text{Dwnstrm(I)} < 158300 - 10^{2}$$

$$158099 < \text{Dwnstrm(I)} < 158200$$

The area with the largest code within this range will be the area immediately downstream of area 158300. The ranges for “even” and “1” highest level non “0” codes are determined in a manner similar to the one above and to the ranges used in the identify subroutine.

As shown in Figure 5.21, the area downstream of area #158300 is area 158100, which is the lowest possible value that lies within the stated range shown above. This lowest value is selected because it is the only value in the database that is within the stated range. This is because the downstream area also happens to be drainage deficient at levels greater than level 4. If the level 4 area 1581 were not drainage deficient at either the level 5 or level 6 scales, then the downstream area for area #158300 would be area #158299. This value represents the highest possible value within the stated search range. However, it is not sufficient to only search the database for values at the extreme edges of the search range. This is obvious when considering the case where the level 4 drainage area is drainage sufficient but the level 5 drainage areas are not. In this scenario, the downstream area for area #158300 would be area 158190, which is in the middle of the stated search range. To account for this scenario, all numbers within the search range must be considered. Further discussions for the use of search ranges are given in Chapter 6.

The second and final function of the density subroutine is to determine the downstream area for areas with codes containing one or more “0” digits, but with a non “0” highest-level digit. According to the Pfafstetter theory, internal basins may be subdivided into higher-level interbasins, basins, and even a single internal basin (Such internal basins will not be included in this discussion because they would have highest level “0” digits). For example, the internal basin in the Western United States theoretically could be divided into areas based upon the rivers that flow into the Great Salt Lake in Utah. The four rivers that drain the largest areas would serve to define the basins, and the interbasins would be the areas that drain directly to the shores of the lake. The numbering of all of these areas would follow the pattern of the continental scale (Level 1) areas. The “2” digit is assigned to the most northern basin (as defined from where the basin enters the internal lake), and the remaining areas are numbered in the clockwise direction.

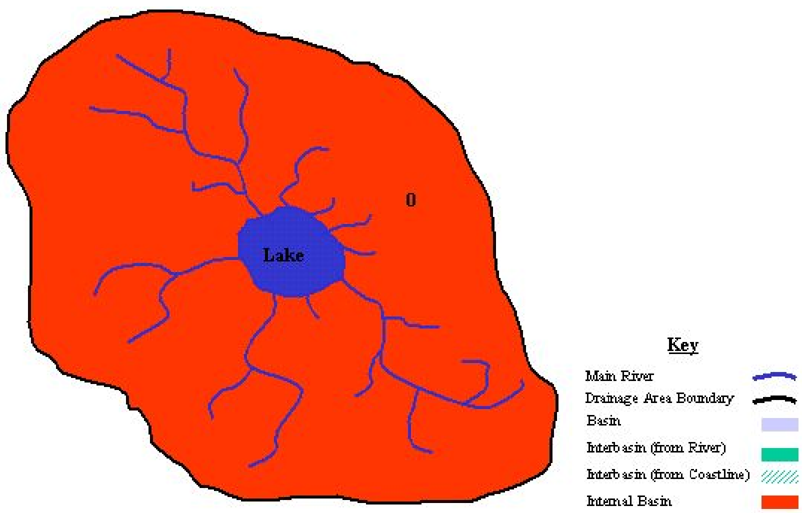


Figure 5.22 – Internal basin surrounding a lake. The area has the “0” code.

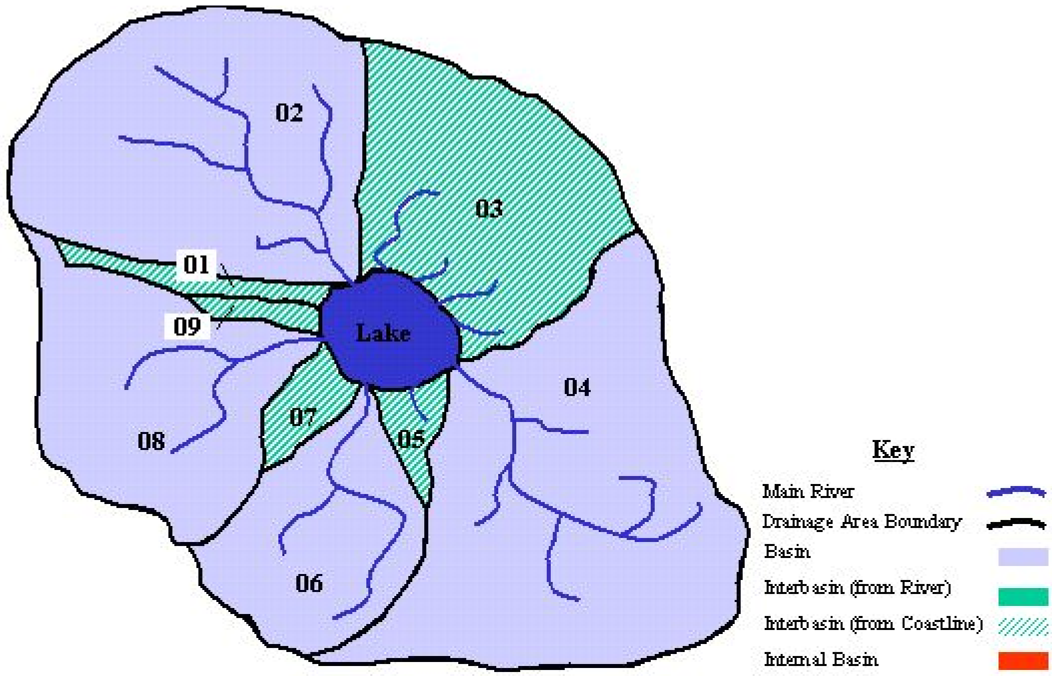


Figure 5.23 – Basins and interbasins draining to an internal sink (lake). Areas are numbered sequentially in the clockwise direction around the coastline in a manner similar to level 1 Pfafstetter code assignment. All areas drain to the sink.

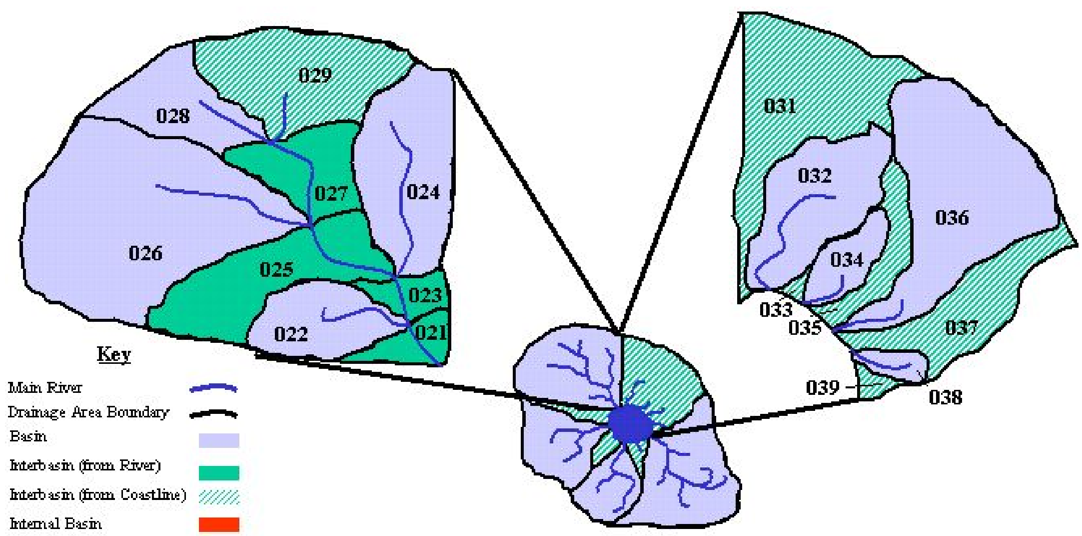


Figure 5.24 – Areas within lower level basins and interbasins draining to an internal sink (lake). The downstream drainages are determined with modified algorithims from the identify and water subroutines.

At the level one higher than the level of the internal basin, all of the interbasins and basins within the internal basin must drain to the internal sink (Figure 5.23). This situation is analogous to the case of level 2 areas derived from level 1 interbasins at the continental scale. At the level two higher than the level of the internal basin, areas within the internal basin will not necessarily drain to the internal sink. For example, areas within interbasins from the next lowest level will always drain to the internal sink and will not have any downstream basins. However, areas within the next lowest level basins will either drain to other areas within the basin or they will drain to the internal sink. (Figure 5.24) The downstream areas for such areas are determinable according to the rules and criteria employed for areas with codes that do not contain "0," specifically the criteria from the water and identify subroutines.

In a general sense, non-drainage deficient areas with "0" digits in their Pfafstetter codes may have downstream areas as long as the highest and penultimate level digits are not equal to "0." For example, area #983026 has two non-"0" digits at higher levels than its "0" fourth-level digit. Using modified versions of the identify subroutine algorithims, the area downstream of this basin is identified as interbasin #983025 (See Figure 5.24). The area downstream of area #983036, however, is the internal sink itself. This is recognized after applying a modified version of criterion #1 for areas draining to the ocean (Section 4.3.4). These modified algorithims apply the downstream criteria of the identify and water subroutines while considering only those digits for levels higher than the highest level "0" digit. Hence, area #983036 is treated as a "pseudo-area" #36, which must drain to an internal sink. Area #983026 is treated as a "pseudo-area" #26 which must drain to a "pseudo-area" #25, so the downstream area is determined to be #983025.

After the density subroutine determines the downstream characteristics of an area, then the Downstream Area Identificaiton process begins again at Stage #1 with the next area entry in the database. However, once all entries in the database have had their downstream characteristics identified, then the final stage of the identification process is initiated. This final stage, Stage #3, involves a form of error-checking on the determinations made in Stage #2. This error checking is very useful for identifying irregularities in the dataset or if the dataset does not contain a complete set of Pfafstetter-based codes.

5.2.5 Stage #3 – The Border and Three_Rivers Subroutines

The downstream area of each database entry is determined with the water, identify, and density subroutines. However, in many instances the downstream areas are determined without assuring that the determined area actually exists in the database. For example, in the identify subroutine, area #9845 is identified as being downstream of area #9846. The determination was made by subtracting "1" from the given area's code. This determination procedure does not assure that an area with the code #9845 exists in the database. However, according to the Pfafstetter-based theory, area #9846 cannot exist in the database if area #9845 does not exist. Therefore, in a dataset with properly attributed and complete Pfafstetter codes, the results of the identify subroutine are unquestionable. Errors in the downstream determination process are possible, however, if the process is applied to a subset of a complete database attributed with Pfafstetter-based codes. The **border** subroutine checks the downstream determination results from Stage #2 and identifies any situation where the identified downstream area does not actually exist within the database.

Pfafstetter codes are best assigned to continental scale datasets, where the landmass described within the dataset is entirely surrounded by water. Such a dataset is really only practical for large-scale hydrologic assessments, whereas local assessments are likely to focus on small sections of the overall continent. It would be impractical to develop an entire Level 6 dataset for North America when the area of interest is only the Continental United States. It is much more practical to perform an analysis on a U.S. subset of the entire North American dataset. A database consisting of only this subset would not include all of the areas within Canada, Alaska, and Mexico that are also included within the North American dataset. Therefore, areas that drain along the Canada-U.S. and Mexico-U.S. borders may not have downstream areas that exist within the specified subset. In such a situation, the downstream area, as identified by the identify or density subroutines, is an area that is not included in the study area dataset.

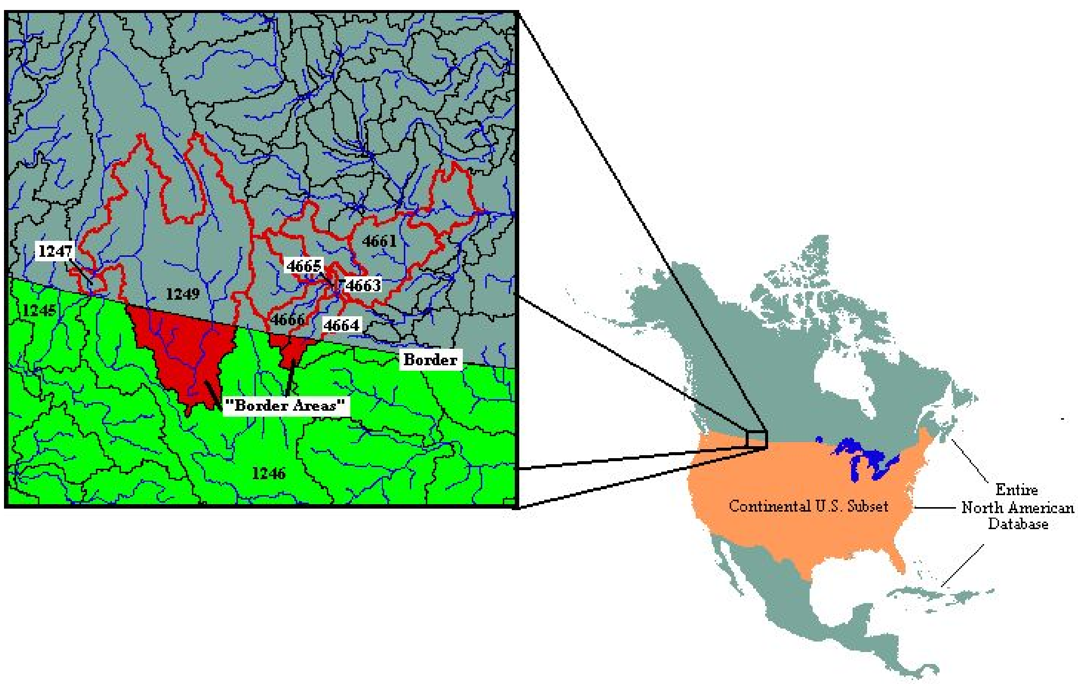


Figure 5.25 – “Border” Drainages on U.S. subset of the North American HYDRO1K dataset.

In determining the downstream areas for the U.S. subset of the North American HYDRO1K dataset, the identify subroutine determines that area #1247 is downstream of area #1249, which spans the U.S.-Canada border (Figure 5.25).

However, because area #1247 is completely outside of the continental U.S., it is not included in the dataset. The border function recognizes that the assigned downstream area #1247 does not exist in the database, and it assigns the Dwnstrm(I) value “-666” and the Comments(I) value “Check - Border” to area #1249. The same occurs for areas #4664 and #4666, which form part of the Hudson Bay drainage system. For these areas, the identify subroutine determines that areas #4663 and #4665 are downstream, respectively. The border subroutine searches the database for areas with these codes, and when they are not found, the subroutine reassigns the Dwnstrm(I) and Comments(I) values. The subroutine functions without regard to the Pfafstetter level, and it ignores all areas with Dwnstrm(I) values less than zero (namely the internal basins, ocean-draining areas, or areas with unrecognizable Pfafstetter codes). The subroutine terminates after each area’s downstream area has had its existence in the dataset either verified or negated.

The final subroutine involved in the Downstream Area Identification process is the **three_rivers** subroutine, which determines downstream relationships in the unusual circumstance that three rivers join at a single confluence. This situation occurs in only one location within the South Eastern section of the Asia HYDRO1K dataset, and it occurs in 5 locations in the North American HYDRO1K dataset. The Pfafstetter numbering system does not account for instances where more than two areas drain to a single confluence. However, in creating the HYDRO1K dataset, researchers at the U.S. Geological Survey adjusted the system to assign codes in these situations. Specifically, in the USGS-Pfafstetter system methodology, two of the areas are assigned basin codes, and the area with the largest upstream drainage receives the interbasin code. This area may receive flow from other areas.

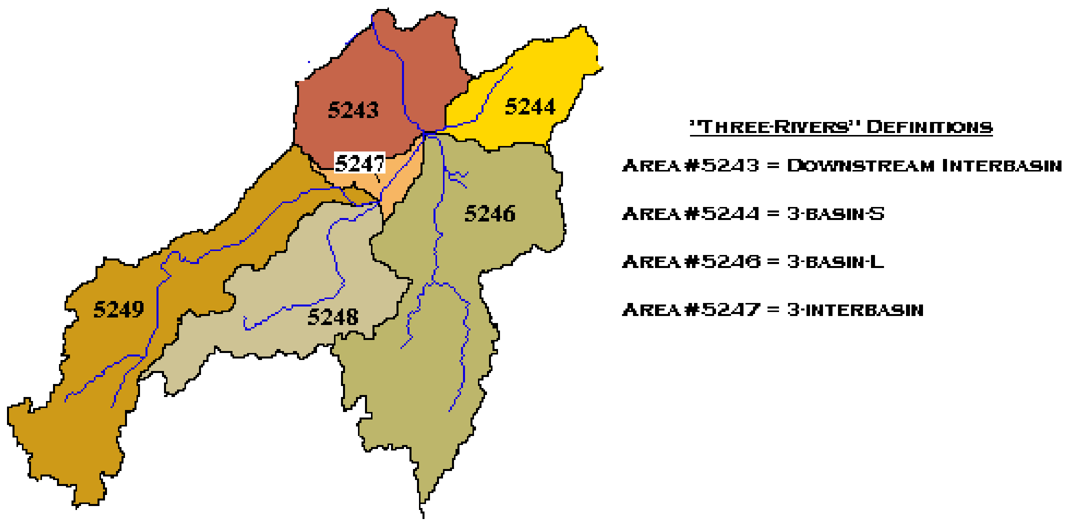


Figure 5.26 – U.S.G.S.-Pfafstetter Coding when three areas/rivers drain to a single confluence (HYDRO1K data for North America)

According to the basic Pfafstetter system for assigning codes to non-coastal areas, two areas drain to a single confluence. The area with the larger upstream drainage area is considered an interbasin, and the area with the smaller upstream drainage area is the basin. The USGS-Pfafstetter system extends this pattern in assigning codes to three or more areas draining to a single confluence. The area with the largest upstream drainage area is considered as the interbasin, and the other areas are basins. For clarity, the areas upstream of the 3-river confluence are noted as 3-interbasin, 3-basin-L, and 3-basin-S where “L” and “S” stand for “large” and “small” and reflect the relative sizes of the basins. The smallest area (3-basin-S) is assigned the lowest Pfafstetter code, which will be equal to the code of the interbasin downstream of the confluence, plus 1. The other basin (3-basin-L) receives the code of the interbasin downstream of the confluence, plus 3. The upstream interbasin (3-interbasin) is then assigned the code 4 greater than the code of the downstream interbasin, which equals a code one greater than that of 3-basin-L. This coding methodology is evident in Figure 5.26.

With this methodology, the missing code is that of an interbasin with the code two-greater than that of the downstream interbasin. This “missing interbasin” would normally receive flow from the 3-interbasin and the 3-basin-L. Therefore, the identify subroutine will assign the code of the “missing interbasin” as the code of the area downstream of the 3-interbasin and 3-basin-L areas. The border subroutine will recognize that this code does not exist within the database, and it will assign the Dwnstrm(I) value “-666” and the Comments(I) value “Check – Border” for these areas. This assignment is made regardless of whether or not the areas in question drain to the border of the study area. The three_rivers subroutine runs on only those areas with the Dwnstrm(I) value “-666,” and its first task is to identify situations where the area in question is on the study area border, and when the area is one of three areas draining to the same confluence.

For instances where three rivers/drainage areas merge at the same confluence, the areas have USGS-Pfafstetter codes corresponding to one of three schemes (Figure 5.27). Each of these schemes requires that the 3-interbasin have a code 4 greater than the interbasin downstream of the confluence. This requires that the 3-interbasin has a highest-level digit code greater than 4. Therefore, any area with a Dwnstrm(I) value of “-666” and with a highest-level digit less than or equal to “4” must be located on the border. In such an instance, the Comments(I) value is changed from “Check – Border” to “Border.” The “Check” is removed because there is no longer any doubt that the area is located along the study area border. The Dwnstrm(I) value in this situation becomes “-777.”

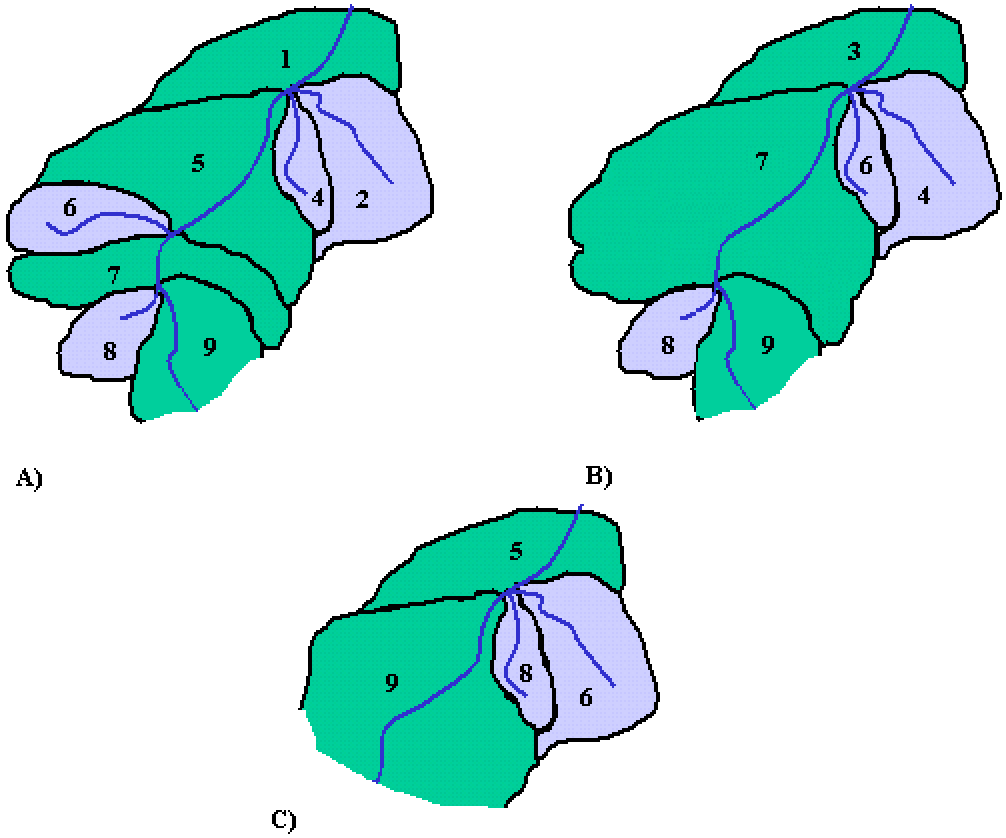


Figure 5.27 – Possible codes for three areas draining to the same confluence (Note the "9" area may or may not have an upstream area)

For an area that has the "-666" Dwnstrm(I) attribute and an odd highest level digit greater than 4, the area may be the 3-interbasin. The three_rivers subroutine searches the database for areas with codes one less, three less, and four less than the area's code. If each of these areas exist in the database, and if the Dwnstrm(I) values of these areas do not equal "-777" (signifying border-draining areas), then the interbasin in question is one of three areas draining to a single confluence. In this instance, the Dwnstrm(I) attribute becomes the code 4 less than the code of the 3-interbasin. The Comments(I) attribute is switched from "Check – Border" to "Success," signifying that the area drains to another area in the database.

However, if any of the desired areas are not included in the database, or if any of them drain to the study area border, then the area must drain to the study area border. In this case, the Dwnstrm(I) value becomes “-777” and the Comments(I) value becomes “Border.”

For an area that has the “-666” Dwnstrm(I) value and an even highest level digit, the area may be the 3-basin-L. The area could not be the 3-basin-S, because this basin has its downstream area correctly determined by the identify subroutine, and it will not be recognized in either the border or three_rivers subroutines. In determining if an area is the 3-basin-L, the three_rivers subroutine searches the database for areas with codes one more, two less, and three less than area’s code. These areas may correspond to the 3-interbasin, the 3-basin-S, and the interbasin downstream of the three-river confluence. If each of these areas exists in the database and if their Dwnstrm(I) values do not equal “-777,” then the basin is the 3-basin-L and its Dwnstrm(I) and Comments(I) are adjusted accordingly. If any of these search areas is not included in the dataset, or if any of the areas has a Dwnstrm(I) value of “-777,” then the basin in question drains to the study area border.

5.2.6 Downstream Area Identification Completion

The Downstream Area Identification macro terminates after running the three_rivers subroutine. At this point, all of the downstream areas have been determined, and the macro writes the Dwnstrm() and Comments() arrays back into the Microsoft Excel spreadsheet as new columns. These columns are named “Downstream” and “Comments,” respectively. A message window appears to announce the macro’s completion and to report the results of the determination (Figure 5.28). The presented results include:

- The number of determinations made
- The number of areas whose drainage was determined correctly

- The number of areas whose drainage needs to be manually determined, and
- The number of areas with unrecognizable codes

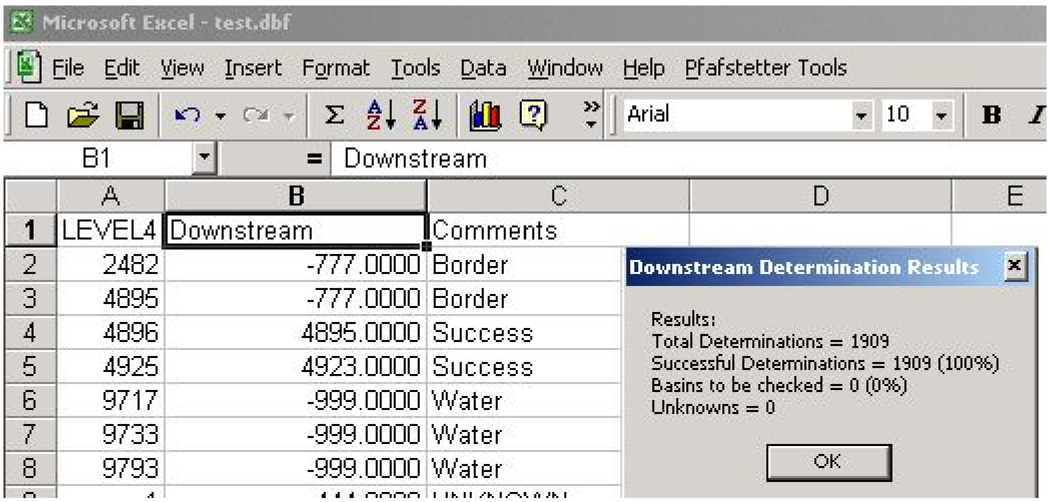


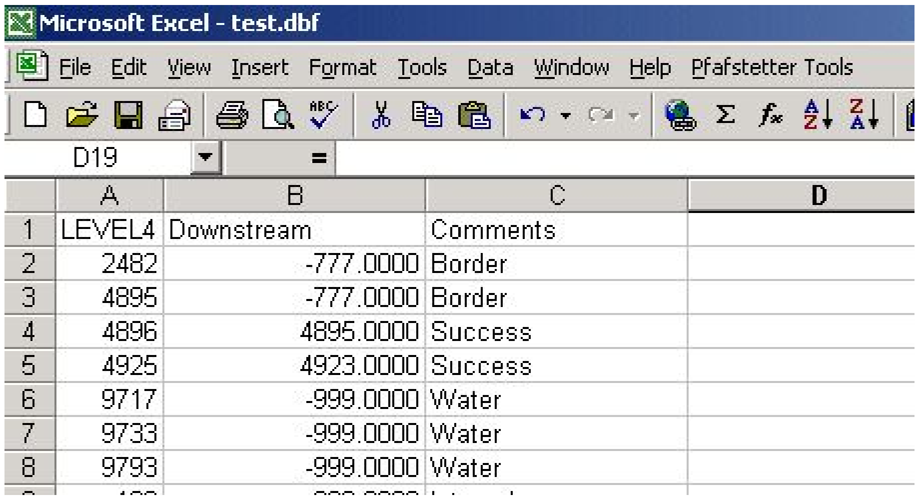
Figure 5.28 -Attribute table after running the Downstream Area Identification macro. A Message window describes the results of the Downstream Area Identification process (HYDRO1K Data for Asia).

5.3 The Upstream Area Identification Macro

The Upstream Area Identification macro determines the areas immediately upstream of each area in a dataset. It uses the methodology described in Chapter 3, and makes upstream determinations based on area ID and downstream area ID values. This methodology is simpler than developing a methodology for determining upstream areas based on topology implied within Pfafstetter codes. However, it is possible to determine upstream areas based entirely on analyses of Pfafstetter-based codes, and basic methodology for these analyses is given in Appendix B.

The Upstream Area Identification macro is designed for use on Pfafstetter-attributed data, but is easily modifiable to work with other datasets (See Section 6.2

for an application to the HUC dataset for the United States). The macro is a stand-alone macro in that it must be called separately from the other macros described in this work. However, it would be a simple matter to combine this function with the Downstream Area Identification macro (Section 5.2) and the Topologic Navigation macro (Section 5.4) to further automate the navigation functionality.



	A	B	C	D
1	LEVEL4	Downstream	Comments	
2	2482	-777.0000	Border	
3	4895	-777.0000	Border	
4	4896	4895.0000	Success	
5	4925	4923.0000	Success	
6	9717	-999.0000	Water	
7	9733	-999.0000	Water	
8	9793	-999.0000	Water	

Figure 5.29 –Attribute table before running the Upstream Area Identification macro. The table must contain the area ID codes (LEVEL4) and the codes of the downstream area (Downstream). (See Figure 5.28)

In order to carryout the Upstream Area Identification macro, the shapefile attribute table must contain at least two fields. One field contains the Pfafstetter codes, and the header for this field must be of the form “LEVELX” where “X” is a positive integer corresponding to the Pfafstetter code level. Capitalization of the letters in “LEVEL” is not necessary. The second required field is the “Downstream” field, which stores the code of the area immediately downstream of area in the database (Figure 5.29). For this discussion, the “Downstream” field is assumed to have been created with the Downstream Area Identification macro (Section 5.2), however the field may be created by other means.

The first step in the Upstream Area Identification process is to determine the maximum number of areas upstream of any single element in the database. This is carried out by determining the single drainage area most often listed as the area downstream of another area in the dataset. The number of times this area is listed as such is counted and stored in the **maxcount** variable. This variable may take on any positive value, but is usually 2 or 3 for Pfafstetter attributed datasets.

Once the "maxcount" is determined, a 2-dimensional **Upstrm()** array is defined, with the number of rows equal to the number of entries in the dataset and its columns equal to the maxcount. The Upstrm() array will store the Pfafstetter codes of the areas upstream of a given area. For example, Upstrm(I,2) stores the 2nd upstream area for the drainage area indexed by I. Calling this area a "second" area implies only that another area has already been determined to be upstream of the selected area. There is not any drainage area dependence between the first upstream area and other upstream areas, and the classification as a "first" upstream area merely states that the program identified that upstream area before identifying others. It would be easy to develop a methodology for assigning "first" and "second" classifications to upstream areas, possibly based on the relative drainage area sizes or other common area attributes. Such a methodology was not developed here because it is not necessary for Topologic Navigation purposes. The upstream areas are determined by analyzing the Pfafstetter codes and Downstream codes of each entry in the dataset. The procedure is shown in Figure 5.30.

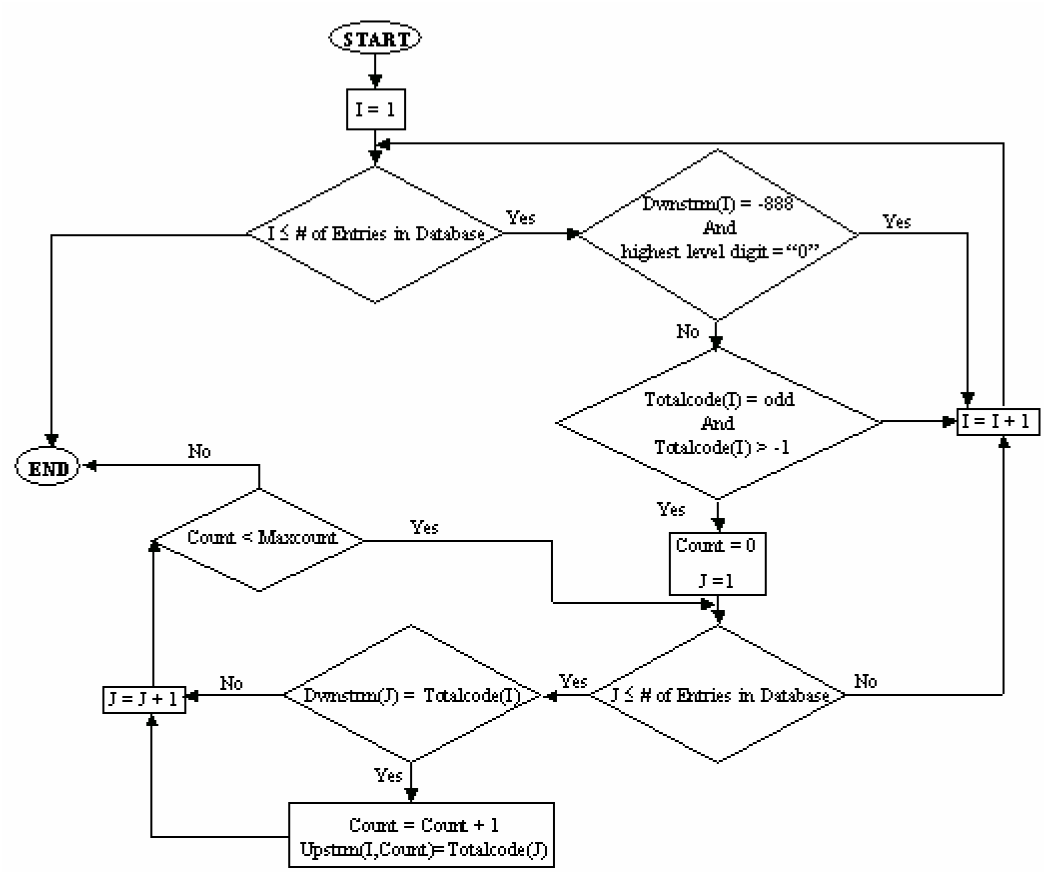
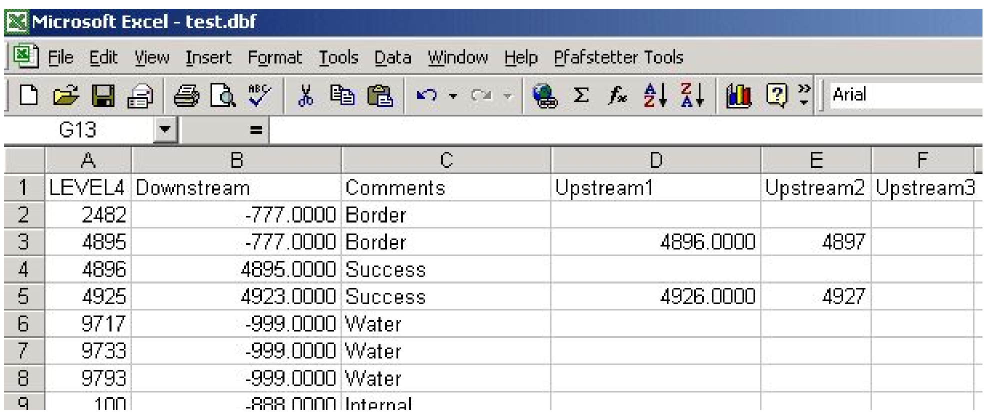


Figure 5.30 – Flow Chart describing Upstream Area Identification Algorithm

In reference to Figure 5.30, the upstream areas are determined by first eliminating from consideration an area if it's Pfafstetter-based code ends in "0" and if it has "-888" as it's downstream attribute. Any area that meets these two conditions must be an internal basin, and cannot have any upstream areas. Both conditions are necessary because of the "drainage-density" problems associated with the HYDRO1K data (Section 5.2). The "-888" downstream code is necessary to separate those "drainage-deficient" areas from true internal basins, since all internal basins have the "-888" downstream code. Next the program considers areas with positive odd Pfafstetter-based codes. An odd Pfafstetter-based code signifies the area is an interbasin, which may have upstream areas. By considering only areas with positive codes, insignificant areas are eliminated from the Upstream Area Identification process (See Section 5.2 for a discussion of negative codes in HYDRO1K data).

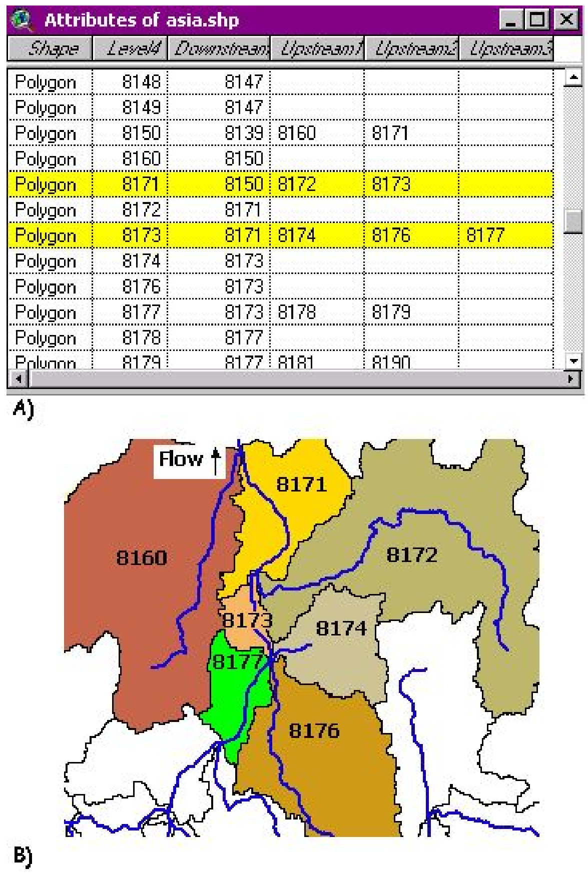
If these conditions are met, the Pfafstetter-based code of the area in question is compared with the codes listed as downstream areas for the database entries. When a match is found, the code of the database entry with the downstream attribute equal to the code of the area in question is stored in the Upstrm() array. The column in which this entry is stored depends on whether other matching entries have already been found. Upon finding a match and storing the appropriate code in the Upstrm() array, the program moves on to the next entry in the database. The program continues until all entries in the database have been considered.



	A	B	C	D	E	F
1	LEVEL4	Downstream	Comments	Upstream1	Upstream2	Upstream3
2	2482	-777.0000	Border			
3	4895	-777.0000	Border	4896.0000	4897	
4	4896	4895.0000	Success			
5	4925	4923.0000	Success	4926.0000	4927	
6	9717	-999.0000	Water			
7	9733	-999.0000	Water			
8	9793	-999.0000	Water			
9	100	-888.0000	Internal			

Figure 5.31 – Upstream Area Identification results in Microsoft Excel. Three “UpstreamX” columns have been added to the dataset from Figure 5.29.

The results of this process are shown in Figure 5.31 for attribute data displayed in Microsoft Excel, and in Figure 5.32 for the same data displayed in ArcView 3.2. The data is part of the HYDRO1K dataset for Asia. Figure 5.32a shows the attribute table for the data, which now contains the upstream element codes. Note how there are three upstream entries for interbasin 8173 but only two entries for interbasin 8171. These basins are shown in Figure 5.32b, where it is obvious that the program performed accurately. Interbasin 8175 does not exist because areas 8177, 8176, and 8174 all drain to the same point, which forms the upstream boundary of interbasin 8173.



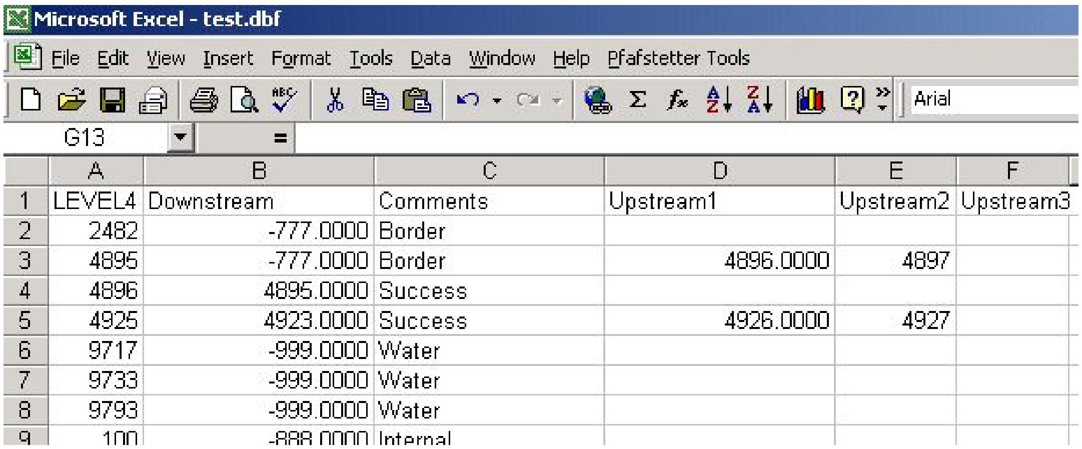
Attributes of asia.shp

Shape	Level4	Downstream	Upstream1	Upstream2	Upstream3
Polygon	8148	8147			
Polygon	8149	8147			
Polygon	8150	8139	8160	8171	
Polygon	8160	8150			
Polygon	8171	8150	8172	8173	
Polygon	8172	8171			
Polygon	8173	8171	8174	8176	8177
Polygon	8174	8173			
Polygon	8176	8173			
Polygon	8177	8173	8178	8179	
Polygon	8178	8177			
Polygon	8179	8177	8181	8190	

Figure 5.32 – Upstream Area Identification Results in ArcView 3.2 - A) Attribute table showing Upstream areas, B) HYDRO1K defined areas with the attributes from A). As determined by the Upstream Element Identifier program, areas 8172 and 8173are upstream of area 8171, and areas 8174, 8176, and 8177 are upstream of area 8173.

5.4 The Topologic Navigation Macro

The Topologic Navigation macro determines the drainage areas upstream and downstream of a user-specified drainage area. The macro is set to run on Pfafstetter attributed data, but is easily modifiable for use on other datasets. An example of such a modification and use is given in Section 6.2, where the macro is run on the HUC dataset for the United States. The macro is a stand-alone function, although it could be modified to run immediately after execution of the Downstream Area Identification and the Upstream Area Identification macros



	A	B	C	D	E	F
1	LEVEL4	Downstream	Comments	Upstream1	Upstream2	Upstream3
2	2482	-777.0000	Border			
3	4895	-777.0000	Border	4896.0000	4897	
4	4896	4895.0000	Success			
5	4925	4923.0000	Success	4926.0000	4927	
6	9717	-999.0000	Water			
7	9733	-999.0000	Water			
8	9793	-999.0000	Water			
9	100	-888.0000	Internal			

Figure 5.33 - Attribute table before running the Hydrographic Navigation macro. The table must contain the Pfafstetter codes (LEVEL4), the codes of the downstream element (Downstream), and the codes of the upstream elements (Upstream1, Upstream2, etc.) Reference Figure 5.32.

The macro is based on the methodology described in Chapter 3, and the attribute table to which it is applied must contain at least three fields. One field contains the Pfafstetter codes, and the header for this field must be of the form “LEVELX” where “X” is a positive integer corresponding to the Pfafstetter code level. Capitalization of the letters in “LEVEL” is not necessary. The second required field is the “Downstream” field, which stores the code of the area immediately downstream of area in the database. This field may be created with the Downstream Area Identification macro (Section 5.2) or it may be created by some other means. The third field must be the “Upstream1” field that stores the code of the area upstream of each area in the database. There may be multiple “UpstreamX” fields, where X is a positive integer. The maximum value of X signifies the maximum number of areas immediately upstream of a given area in the database. For areas attributed with Pfafstetter-based codes, X is usually either equal to “2” or it is equal to “3.” These fields contain the Pfafstetter codes of the areas immediately upstream of each entry in the database. The “UpstreamX” fields may be created with the Upstream Area Identification macro (Section 5.3) or by some other means.

Upon running the macro, the user is prompted to enter the relevant Pfafstetter level, and then is prompted to enter the code of the drainage area from which the navigation is to be performed. This area becomes the initial target area from which the navigation is to be performed (See Chapter 3). The macro then checks to see if such a code exists in the database. If the code does not exist, the user is prompted to enter another code or to cancel the program. If the code does exist, the program continues and the navigation is initiated.

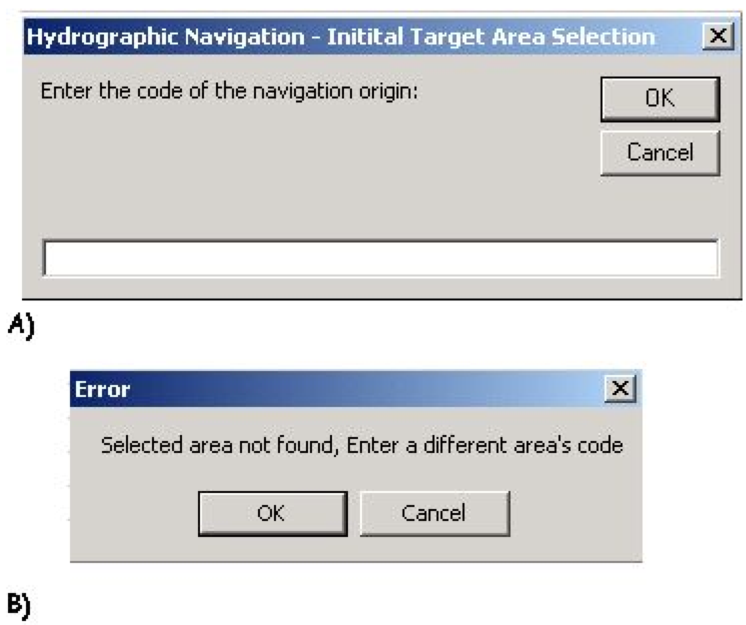


Figure 5.34 – User prompts in the Topologic Navigation macro – A) Initial target area selection, B) Error prompt – the selected area does not exist in the database.

The navigation is conducted in two parts, with the first part tracing downstream from the initial target area and the next part tracing upstream from the initial target area. In order to run each part of the trace, the source dataset must have the appropriate upstream and downstream attributes. If these attributes do not exist, the navigation function will not run. However, if upstream attributes exist without downstream attributes (or vice versa), the function will conduct the navigation using whichever attribute set is available. Assuming both the downstream and upstream attributes exist, the downstream trace is performed first using the **trace_down** subroutine. The trace_down subroutine is shown in Figure 5.35. The subroutine uses two indices to reference attribute data from the database arrays. The **test** index is the number of the array element storing the target area attributes, and the **I** index is the number of an array element in the database. Both indices may take on values of 1 to the number of entries in the database. When an area is found to be downstream of the target area, this downstream area is assigned the **Trace(I)** attribute "2." This attribute was arbitrarily chosen to indicate that the area is downstream of the user-selected area.

The subroutine functions by considering the downstream attribute of its target area, which initially is the user-selected area. If this attribute is negative, then the target area is either an internal basin, or it drains to the ocean or to the study area boundary. Such an area does not have any downstream elements, and the program displays a message saying the initial target area lacks any downstream areas. After the message, the trace_down subroutine ends. However, if the downstream attribute is positive, then there is a downstream area and the navigation continues.

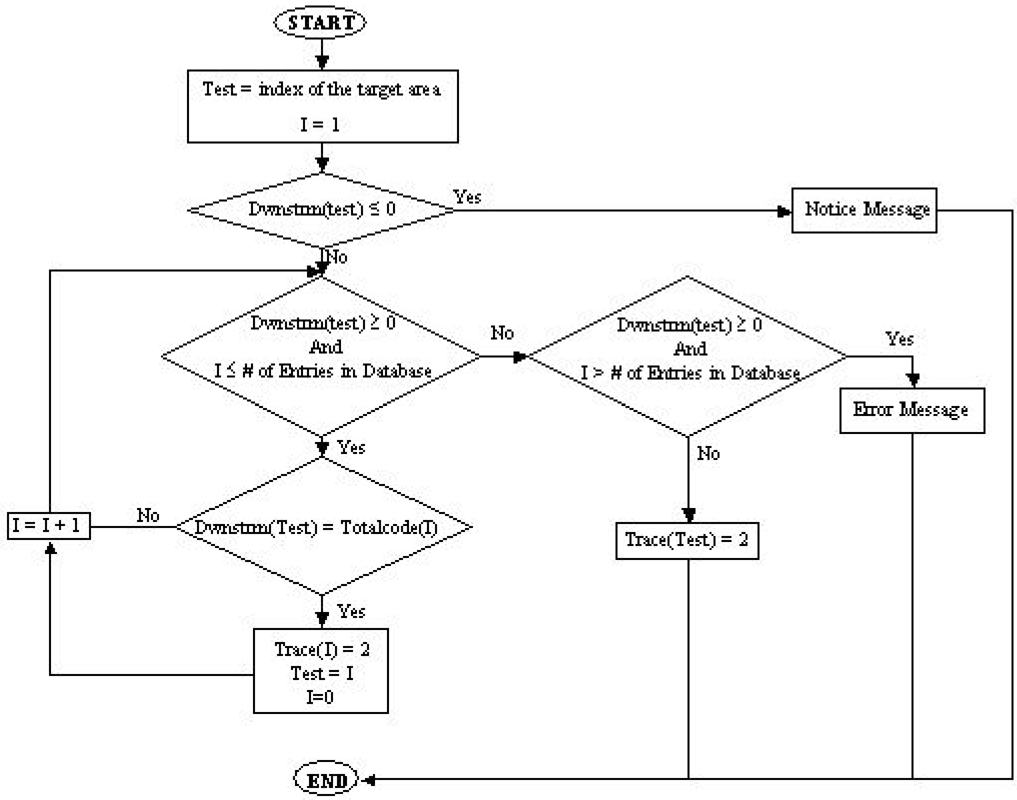


Figure 5.35 – Trace_down Sub-routine in the Topologic Navigation macro (Schematic View)

The next step is to determine if the target area has a downstream area <u>and</u> if the **I** index is less than the number of entries in the database. If the initial target area has a downstream area, then on the first pass through the subroutine loop, both of these conditions will always be met. However, with further passes through the subroutine loop, eventually one of the two conditions will not be satisfied. In such a situation, the navigation ends. It is necessary to include the Dwnstrm(I) ≥ 0 condition at this second step because the notice message associated with the failure of this condition at the previous step is only appropriate for the initial target area. Also, when this condition is not met for a non-initial target area, the area is assigned the Trace(I) attribute "2" signifying it is downstream of the initial target area. By treating the initial target area separately from later target areas, the program avoids assigning the “2” Trace(I) attribute to the initial target area. The initial target area may not have this attribute, because this would mean the area is downstream of itself.

The navigation is performed by comparing the downstream attribute of the target area with the Pfafstetter code of each area in the database. The match is found by incrementing the index **I** until the downstream attribute of the target area equals the Pfafstetter code of an entry in the database. In equation form, this is given as:

$$\text{Dwnstrm(test)} = \text{Totalcode(I)}$$

When this equation is true, the Trace(I) attribute is assigned the value “2”. The target area becomes the Ith area, so that the next phase of the trace will determine the area downstream of the area already determined to be downstream of the initial target area. This progression of target areas is shown in Figure 3.6. However, before comparisons are made with the new target area, the index I is set to 0 so that the next downstream trace considers all entries in the database. As shown in Figure 5.35, after a match is found, I is incremented before comparisons are made with the new target area. Therefore, the first comparison is made with I = 1, signifying that the first entry in the database is the first entry considered in the subroutine loop.

This process of finding the area downstream of a target area, labeling the downstream area, and then making this area the next target area repeats until the downstream attribute of the target area is negative, or until a dwnstrm(test)-totalcode(I) match is not found in the database. Therefore, at this point there are two possible reasons for the downstream navigation to terminate.

The first, and most likely reason for trace_down termination occurs when the dwnstrm(test) value is negative. This signifies that the target area does not drain to another area in the database. The most likely possibility is that the target area drains directly to the ocean, however the area might drain to an area outside of the study area included in the database. For this specific situation, the navigation function must be run on a subset of a larger database containing Pfafstetter attributed data (See Section 5.2). A third, and unlikely, possibility is that the target area drains to an internal waterbody. This possibility is also discussed in depth in Section 5.2. In each of these situations, it is impossible to navigate further downstream, and the trace_down subroutine terminates. Before doing so, however, the target area is assigned the Trace(I) attribute “2” to signify that it is downstream of the initial target area.

The second reason for the subroutine termination is that comparisons have been made between the downstream attribute of the target area and each area in the database, without a match being found. This occurs when the index value I becomes greater than the number of entries in the database. If this situation arises, the program displays an error message informing the user that the downstream areas were determined incorrectly. As discussed in Section 5.2, the Downstream Area Identification macro assigns downstream area IDs based on the entries in the database, and cannot assign non-negative downstream attributes that do not exist as Pfafstetter-based codes in the database. Therefore, if a certain positive attribute exists in the “Downstream” field and not in the “LEVELX” field, then the attribute was assigned incorrectly by some method other than through the use of the Downstream Area Identification macro. One possible exception to this rule is the situation in which downstream attributes were assigned to an entire database, and the navigation function is executed on a subset of this complete database. If it is determined that the downstream areas were determined incorrectly, then the trace_down subroutine terminates without clearing the assigned Trace(I) attributes. It is possible that the areas with the Trace(I) “2” attribute were assigned correctly, and that the trace was successful up until the last area. The user must check the trace results to determine if this is the case.

Upon completion of the trace_down subroutine, the **trace_up** subroutine is run. This subroutine follows the same basic procedure as the trace_down subroutine, namely identifying the upstream area and using this area as the target area for the next upstream determination. However, the subroutine is more complex because multiple upstream areas may exist for each single target area. For example, if each area has two upstream areas, then the number of new target areas increases to 2^X at the X^{th} upstream iteration.

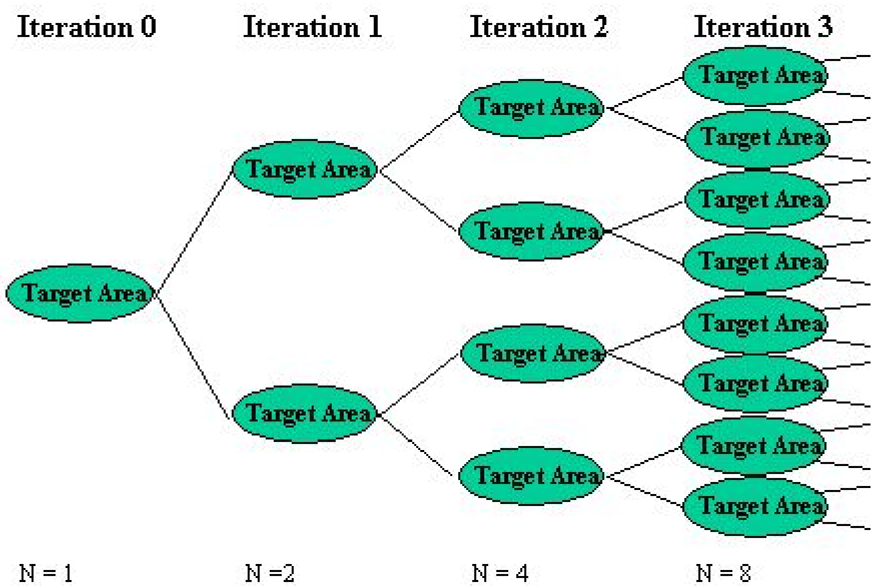


Figure 5.36 – Exponential Growth of Upstream Target Areas
(N = number of target areas)

The Trace_up subroutine handles this multiple-target area problem by storing referenced information in two arrays – specifically the **Checked(I)** array and the **Check1(I)** array. Both of these arrays contain the same number of elements as there are entries in the database. The Check1(I) array stores a "1" value for a target area if that area has upstream areas. In the Trace_up subroutine, the **test1** index is used to reference the initial target area. If it has an upstream area, then check1(test1) = 1. This flags this area as one for which the upstream areas must be determined. The Checked(I) array stores information as to whether the upstream areas of a target area have already been determined. The array is initially set to "NO" for all entries, and an entry is changed to "YES" when the entry has had its upstream areas determined.

The subroutine will determine upstream areas only for I indices with Check1(I) = 1 and Checked(I) = "No". The methodology is then to identify the upstream areas for each target area, give them the "1" attribute in the Check1(I) array, and to continuously cycle through the Check1(I) array, using each area with a "1" attribute as a target area for determining upstream areas. For each of these target areas, once their upstream areas are determined, the Checked(I) attribute becomes "Yes." When the Checked(I) attribute is "Yes", the program recognizes that this area has already been a target area and therefore has already had its upstream areas determined. The program then increments the index value (I), and considers the next entry in the database. The program will end when the index value I is greater than the number of entries in the database. This signifies that all of the areas with the Check1(I) attribute "1" also have the Checked(I) attribute "Yes", therefore that all upstream areas have been determined. At this point, the subroutine terminates.

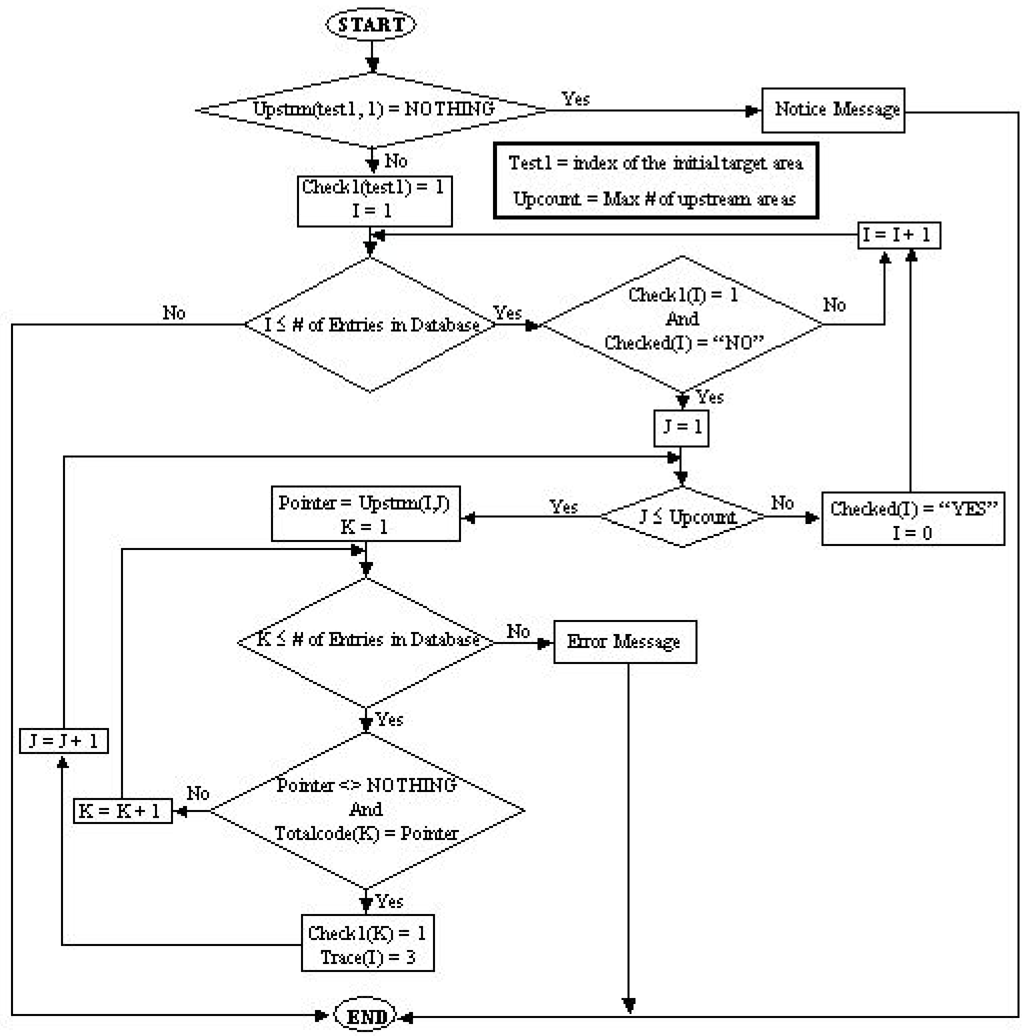
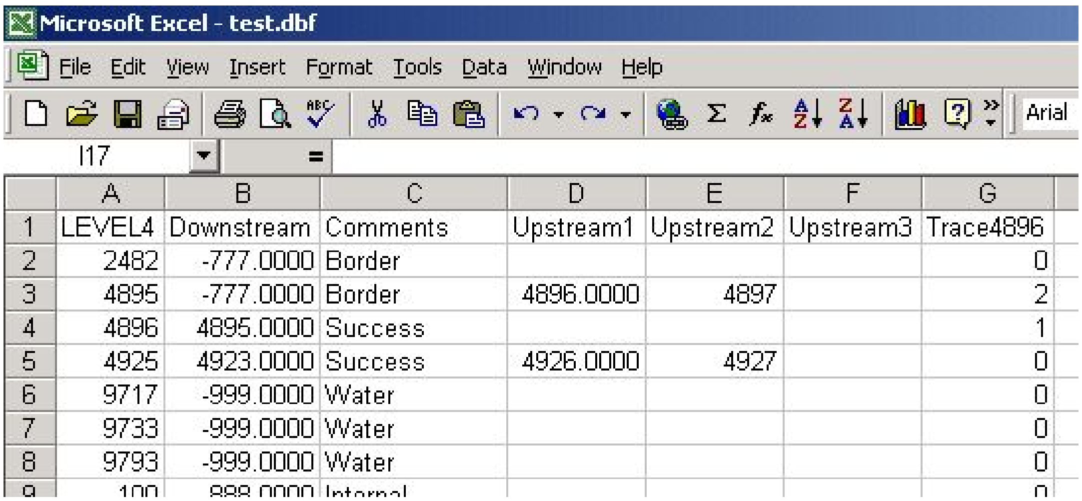


Figure 5.37 - Trace_up Sub-routine in the Topologic Navigation macro (Schematic View)

In the trace_up subroutine, if an area is identified as being upstream of a given area, it is given the code "3" as its Trace(I) attribute. This "3" value was arbitrarily chosen to distinguish upstream areas from downstream areas, from the initial target area, and the other areas in the database that are not topologically related to the initial target area. Upon macro completion, the initial target area is assigned the Trace(I) attribute "1". All areas not identified as being upstream of, downstream of, or equal to the initial target area are given the Trace(I) attribute

"0". Therefore, all areas in the dataset are assigned a Trace(I) value, and the dataset may be displayed or queried based on this value. This clearly and visibly identifies each of the 4 area types resulting from the navigation.

The results of this Topologic Navigation process are shown in Figure 5.38 for attribute data displayed in Microsoft Excel, and in Figure 5.39 for U.S. HYDRO1K data displayed in ArcView 3.2.



	A	B	C	D	E	F	G
1	LEVEL4	Downstream	Comments	Upstream1	Upstream2	Upstream3	Trace4896
2	2482	-777.0000	Border				0
3	4895	-777.0000	Border	4896.0000	4897		2
4	4896	4895.0000	Success				1
5	4925	4923.0000	Success	4926.0000	4927		0
6	9717	-999.0000	Water				0
7	9733	-999.0000	Water				0
8	9793	-999.0000	Water				0

Figure 5.38 – Topologic Navigation results in Microsoft Excel. The field "Trace4896" has been added to the attribute table shown in Figure 5.33. The new field name contains the number "4896" because area "4896" was the initial target area.

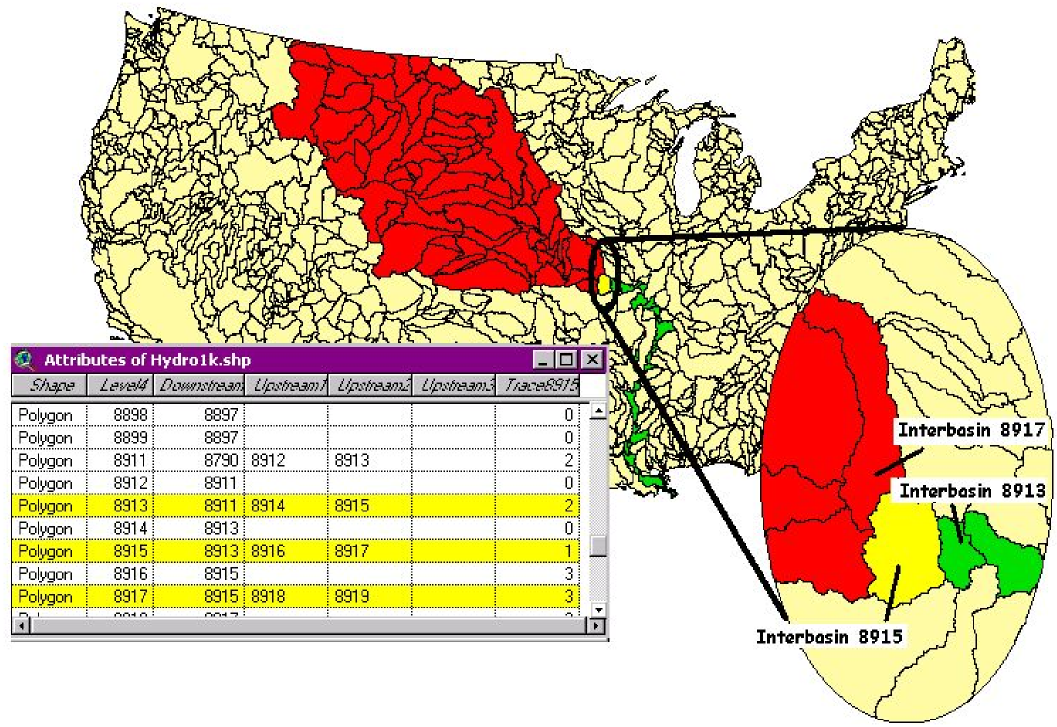


Figure 5.39 – Topologic Navigation from Interbasin 8915 – US HYDRO1K data. Red areas are upstream of interbasin 8195 (Yellow). Green areas are downstream of interbasin 8915. This trace includes the majority of the Missouri River basin and the Mississippi River.

5.5 The Pfafstetter Tools – Navigation Support

The macros described in Sections 5.2-5.4 are necessary for applying the Topologic Navigation methodology to data already attributed according to the Pfafstetter System. These macros are just three of the eight macros that would be useful in a more complete version of the Pfafstetter Tools set. The remaining 5 macros would serve to support and organize the database for use with the Topologic Navigation methodology. These tools are currently underdevelopment and are explained fully in the companion report for this work, entitled “Hydrologic Data Development for South East Asia and Latin America” (Furnans and Olivera, 2002). This section provides a brief overview of the functionality of each of the 5 remaining Pfafstetter Tools:

Pfafstetter Tools

- Downstream Area Identification
- Upstream Area Identification
- Topologic Navigation
- **Node Swap**
- **Stream Filter**
- **Code Assignment**
- **Downstream Element Name**
- **Upstream Element Name**

The **Node Swap**, **Stream Filter**, and **Code Assignment** macros work together to assign Pfafstetter codes to any set of hydrographic stream data. The **Downstream Element Name** and **Upstream Element Name** macros provide the data user with more information about the elements downstream and upstream of each element in the database. All of these functions are designed for use in conjunction with the ESRI ArcGIS and ArcView software and are being written as macros in Microsoft Excel.

5.5.1 The Node Swap Macro

The **Node Swap** macro determines the proper upstream and downstream connectivity of arcs within a river network shapefile. Any given arc is bounded by nodes, referred to as the “TNODE” and the “FNODE.” In the case of arcs that represent rivers, water flows along the arc from the “FNODE” to the “TNODE.” However, if coastlines are represented within the arc shapefile, this “FNODE” to “TNODE” flow connectivity is not always properly defined. The Node Swap macro checks the nodes of each arc in the database and assures that the “FNODE” is always downstream of the “TNODE” for non-coastline arcs. If this is not the case, than the function switches the “FNODE” and “TNODE” for the arc in question. For the coastline arcs, the “FNODE” to “TNODE” connectivity is checked to assure that each coastline node is the FNODE for one and only one coastline arc, and that each coastline node is the TNODE for one and only one coastline arc. If each coastline arc does not have a unique TNODE and FNODE, then at least one of the coastal nodes is a point of divergence or convergence for a mythical flow along the coastline. The Node Swap macro adjusts the coastline nodes in order to assure that the coastline is continuously navigable from one arc to another.

The Node Swap macro is useful because it establishes the topologic relationships between arcs in the database. This relationship is vital in the process of assigning Pfafstetter-based codes.

5.5.2 The Stream Filter Macro

The **Stream Filter** macro makes use of the topological relationships defined by the Node Swap macro to develop a one-to-one relationship between arcs and watersheds in an area. This function is run on an arc shapefile, where each arc carries the identification number of the drainage area containing the arc. There may be hundreds of arcs within a single drainage area, and many of these arcs are not necessary for determining the upstream and downstream connectivity of the drainage area. The stream filter function identifies those arcs that are necessary for determining the connectivity of drainage areas from those arcs that are extraneous. The filtered arcs are often easier to process than the entire set of arcs within the original database.

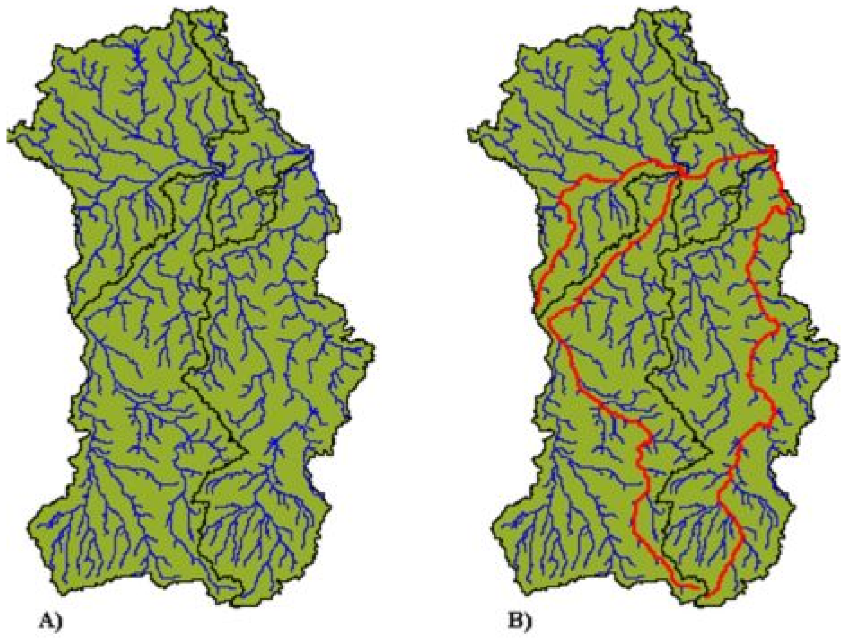


Figure 5.40 – Stream Filtering – A) Original Streams, B) Filtered Streams (Red) among original streams

The stream filter process identifies the main rivers in an interbasin and identifies the longest river within a basin. For coastal interbasins, the longest set of arcs that drains to the coastline is selected. The coastline itself is also selected. The topology implied by these selected arcs is then sufficient to assign Pfafstetter codes.

5.5.3 The Code Assignment Macro

Although the Topologic Navigation macro is complete, it will not gain widespread use because most watershed datasets are not attributed with Pfafstetter codes or downstream areas codes. A method for assigning Pfafstetter codes needs to be developed and tested in order to expand the usefulness of the Topologic Navigation technique. This method should incorporate all of the modifications discussed in Section 6.5, and should produce results without reference to the geographical extent of the study area in question. However, before such a method is developed, a consensus must be reach among all interested parties on the properties of the new Pfafstetter-based system.

Researchers at the EROS Data Center of the United States Geological Survey have developed an Arc Macro Language (AML) program that will assign Pfafstetter codes on geographic data. This program makes use of flow accumulation grids to assign drainage area codes (Verdin, 2001). The exact procedure for assigning the codes has not been published, however the technique is tied into the older versions of GIS technology, specifically ArcView and ArcInfo. AML programs are being phased-out of use in GIS programs with the introduction of ArcGIS by ESRI, inc. The ArcGIS program may be customized with Visual Basic™ programming code, and not the AML programs that were developed with the older GIS software packages. Therefore, it is necessary to write a program in the visual basic computer language that will assign Pfafstetter-based codes to drainage areas. It might be feasible to transcribe the EROS Data Center's AML into visual basic for use in the ArcGIS system, however it would also be useful to develop a code assignment program that only requires vector data.

The **Code Assignment** macro that, upon completion, will be included within the Pfafstetter Tools is a complex function that assigns Pfafstetter codes to vector data. Specifically, it assigns codes to the arcs identified in the stream filter process. These codes may then be transferred to the areas surrounding the arcs by using the query and join functions within ArcView and ArcGIS. This macro requires that the each arc in the dataset is attributed with the ID of the surrounding drainage area, and with the area (in length squared units) of that drainage area. The drainage area ID may be any numerical value, and it will be replaced by an appropriate Pfafstetter-based code upon completion of the macro. The macro calculates the total upstream drainage area for each node within the dataset, and uses the topology implied by these nodes to assign the Pfafstetter codes.

With this function, Pfafstetter codes are quickly assigned to a set of watersheds with drainages defined according to a river network shapefile theme. This function may therefore be used to assign codes to datasets other than the HYDRO1K dataset, and therefore it expands the applicability of the Topologic Navigation methodology and macro.

5.5.4 The Name Identification Functions

The final two macros that make up the Pfafstetter Tools are name identification functions. These macros are only useful with datasets whose elements each have unique names (Other than their IDs). Specifically, the **Downstream Element Name** function determines the name of the area immediately downstream of each element in the dataset. The **Upstream Element Name** determines the names of the areas immediately upstream of each element in the dataset. These functions use the output from the Downstream Area Identification and Upstream Area Identification macros, respectively. They attribute each area with the name of the area(s) immediately downstream or upstream. These functions are useful for regulatory purposes so that the data user can report that the “Hill Country Basin” is upstream of the “West Lake Interbasin,” which may be more useful than reporting that area #9866 is upstream of area #9865.

Chapter 6 – Results and Discussion

6.1 Navigation on Pfafstetter Attributed Data

Currently, only the HYDRO1K dataset contains drainage areas attributed according to a Pfafstetter-based system. The Downstream Area Identification, Upstream Area Identification, and Topologic Navigation macros were applied to various HYDRO1K datasets in order to test their accuracy and applicability. The results of these tests show that the macros perform correctly, although some modifications to the Downstream Area Identification macro are necessary with regard to internal basins and areas with greater than two upstream drainage areas. Also, the test results demonstrated further inconsistencies between the Pfafstetter and USGS-Pfafstetter systems.

Out of the 3,231 level 4 drainage areas on the North American Continent in the HYDRO1K dataset, 23 areas had their downstream characteristics incorrectly determined by the Downstream Area Identification macro (Figure 6.1). Of these 23 areas, eleven were designated as areas draining to the study area border, which is an impossibility because the macro was applied on the continental scale rather than on a subset of USGS-Pfafstetter attributed data.

These eleven areas have codes for which the downstream area as identified by the **identify** subroutine does not exist within the database. Six of these areas were so identified because they are areas upstream of a confluence at which three rivers meet. This situation is easily recognizable in the **three_rivers** subroutine, as long as the areas involved are not drainage-deficient areas. In the HYDRO1K dataset, there are three locations at which 3 level 3 drainage areas each are upstream of a single level 3 interbasin. These areas are drainage deficient, and at the level 4 scale they are attributed with a “0” level 4 digit. As such, their

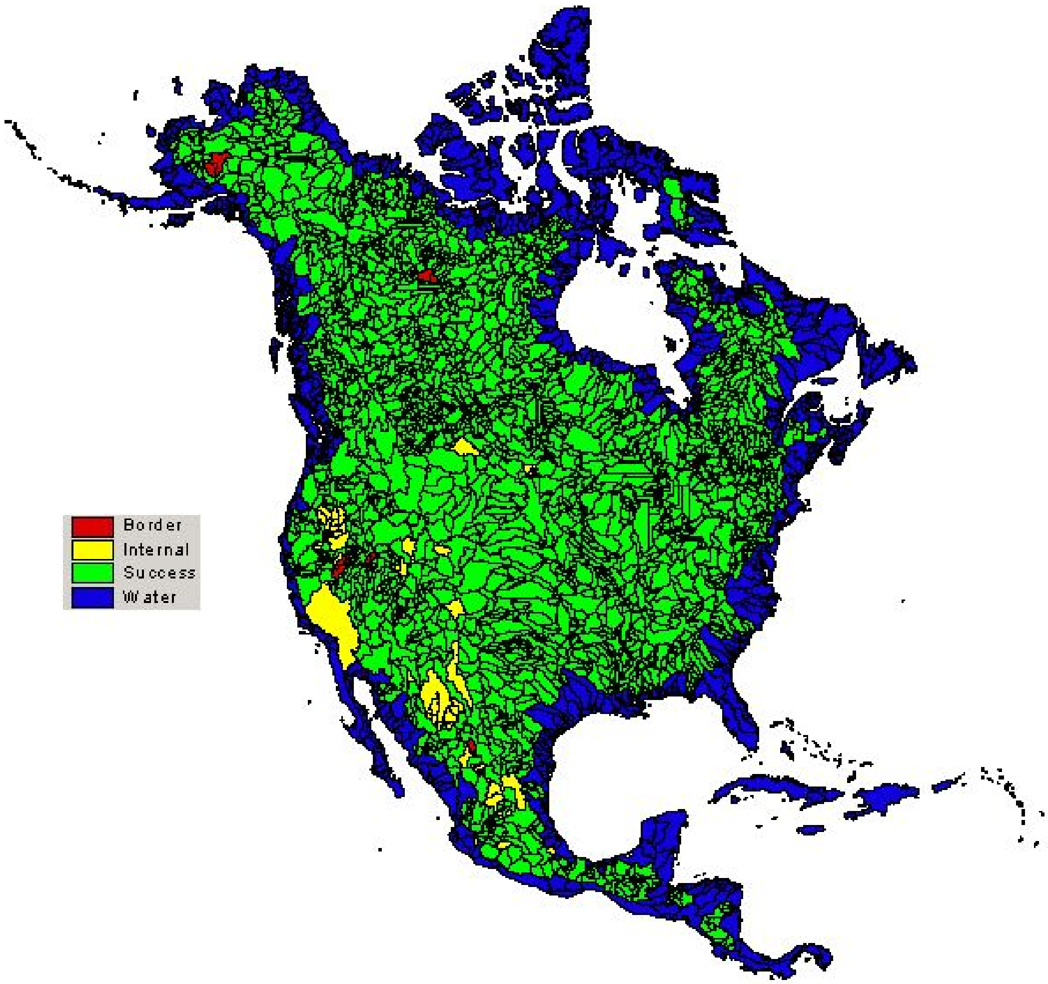


Figure 6.1 – Downstream Area Identification Results for North America, categorized by the "Comments" field. (HYDRO1K Data)

downstream areas are determined by the density subroutine, which assigns a downstream area based on an applicable search range. This search ranged is developed by considering the implied topology of lower level Pfafstetter areas, and it assumes these lower level areas fit the standard format of two areas upstream of a non-coastal interbasin. The search range does not consider the possibility that the lower level areas might be part of a "three_river" type drainage system.

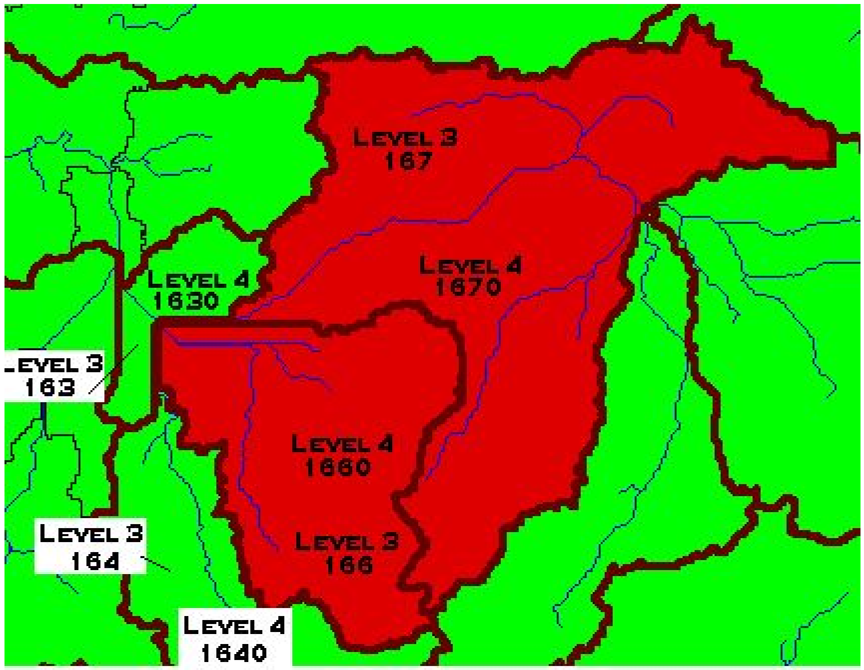


Figure 6.2 – Incorrect Border determinations due to lower level "three_river" situations

As shown in Figure 6.2, at the level 3 scale, areas #166 and #167 are correctly identified as draining to area #163. This identification is made with the three_rivers subroutine, and area #164 is the smaller of the two basins upstream of area #163. However, at the level 4 scale, areas #1660 and #1670 drain into area #1630, yet areas #1660 and #1670 are incorrectly given the "Border" downstream characteristics. This failure of the Downstream Area Identification macro stems from a combination of two the differences between the Pfafstetter and the USGS-Pfafstetter coding systems. The first difference is that the USGS-Pfafstetter system considers situations in which three drainage areas merge at a single point. The second difference is that the USGS-Pfafstetter system assigns "0" digits to drainage deficient areas. The Downstream Area Identification macro is capable of handling each of these differences individually in order to produce correct results, but it was not designed to handle both differences at the same instant. The Downstream Area

Identification macro needs to be modified in order to avoid such determination errors. Also, with modifications to the USGS-Pfafstetter system, these errors become avoidable (Section 6.5).

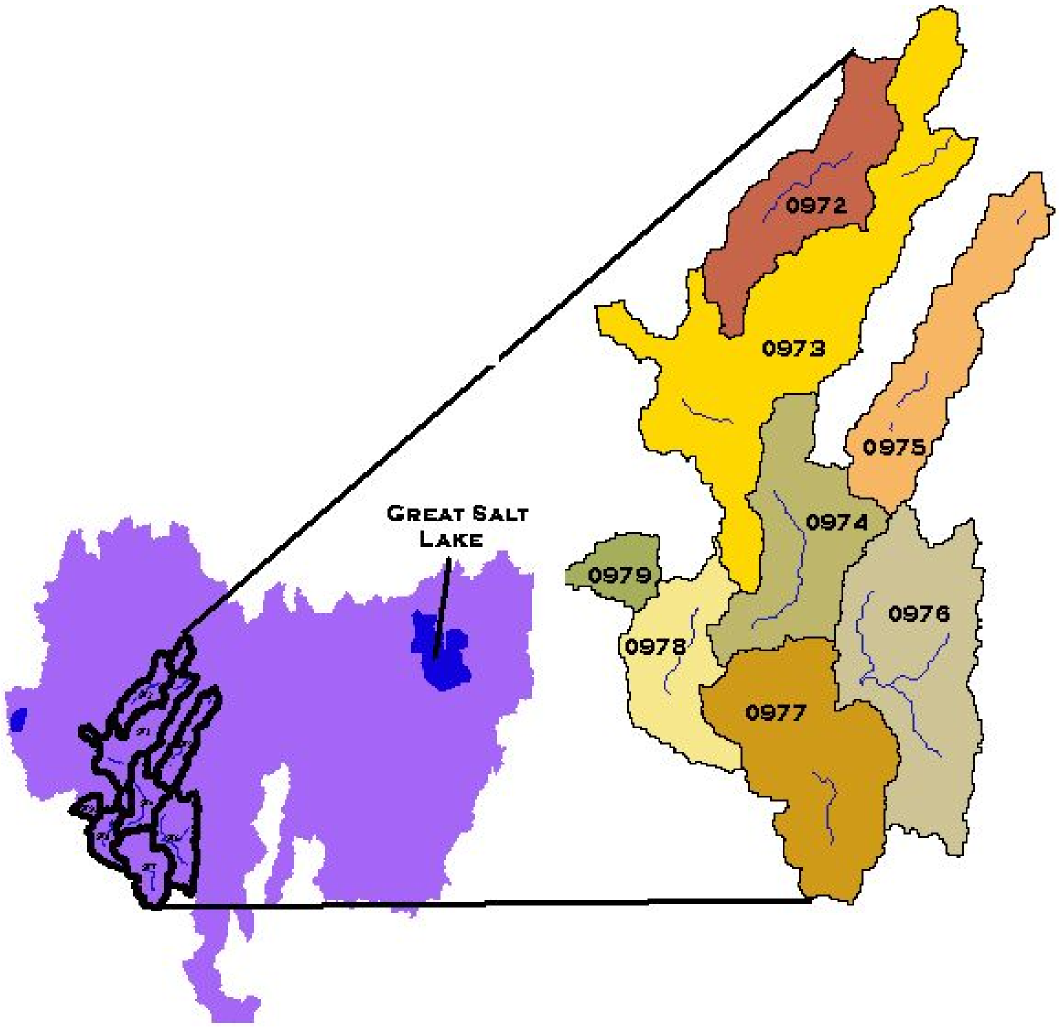


Figure 6.3 – Incorrect Implied Topology For Internal Basins (USGS-Pfafstetter system).

The remaining 5 areas with the “Border” characteristics are all internal basins. The USGS-Pfafstetter system methodology for assigning codes to internal basins in HYDRO1K data makes use of the geographic location of the basins at the expense of the topology defined by the drainage system. As in Figure 6.3, the codes for internal basins within the Great Salt Lake internal basin incorrectly imply that the areas drain to one another. At the Level 1 scale, all of these areas are assigned the “0” digit. The level 2 digits each identify separate distinct internal basins within the level 1 area. The level 3 and level 4 digits serve to further divide the level 2 internal areas. Yet by assigning these areas with the digits 1-9, a drainage system is implied. Area #0975 should drain to area #0973, and area #0978 should drain to area #0977. This is how the **density** subroutine would determine the downstream drainage. The Downstream Area Identification macro considers the determination process for areas #0974-#0979 to be a “Success” because the downstream codes as determined by the density subroutine correspond to areas that actually exist within the database. This accounts for the 12 “non-border errors” from the Downstream Area Identification process. Area #0971 should be downstream of areas #0972 and #0973, but because area #0971 does not exist within the database, areas #0972 and #0973 are assigned the “Border” downstream attributes. The remaining 5 border errors were derived from this use of geographic location in assigning internal basins codes in the USGS-Pfafstetter system. This is inconsistent with the Pfafstetter system methodology that does not require the use of geographic knowledge is assigning codes to areas.

This analysis suggests that the methodology for assigning codes to internal basins needs to be revamped and improved. Internal basins do not fit the drainage pattern to which the Pfafstetter system is easily applicable, and as such the USGS-Pfafstetter system had to include modifications to the Pfafstetter theory in order for use on HYDRO1K data. These modifications are inconsistent with the goals of Pfafstetter theory, and further modifications are necessary with respect to the drainage density problem and the implied topology problem identified herein. Possible solutions to these internal numbering problems are discussed within Section 6.5. With the current HYDRO1K dataset, it is best to visually determine the drainage relationships for areas containing a “0” digit at any level other than the highest level.

Despite these downstream determination failures, the topology of most drainage areas is easily determinable with the Pfafstetter Tools. As shown in Figure 6.1, the Downstream Area Identification macro correctly identifies all of those areas that drain directly to the ocean. The Topologic Navigation methodology is viable based on the results of the Downstream Area Identification macro, the Upstream Area Determination macro, and the Topologic Navigation macro. (Figure 6.4).

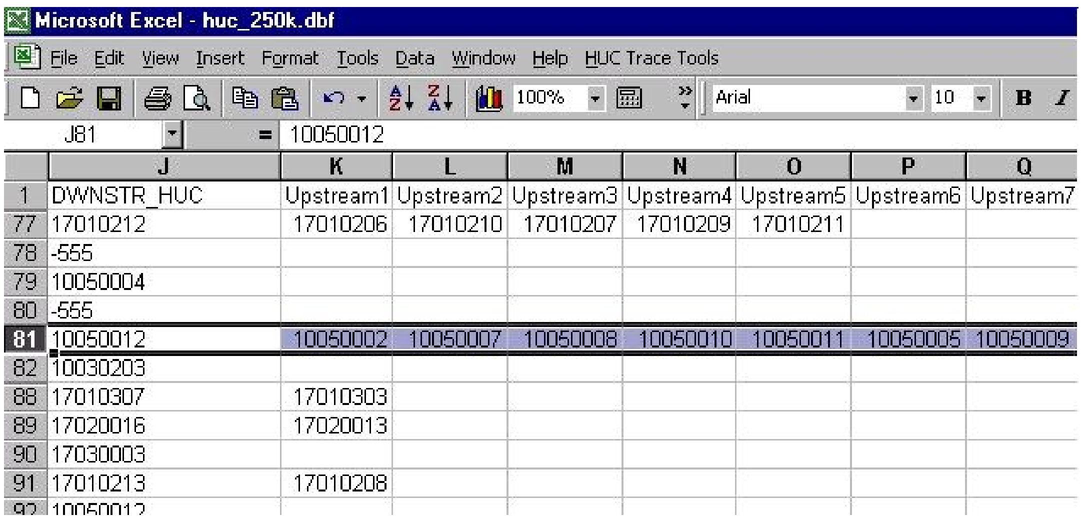
Figure 6.4 – Hydrographic Navigation on HYDRO1K Data (North America) – 80 areas are upstream of area #8911, which flows into the Mississippi River drainage system.

6.2 Navigation on Non-Pfafstetter Attributed Data – U.S. HUCS

One key aspect of the Topologic Navigation methodology that separates it from the tracing technique developed by Verdin (2001) is that it is applicable to any type of data that includes connectivity information. The Upstream Area Identification and the Topologic Navigation macros are not dependent upon use of any Pfafstetter-based coding system. The Upstream Area Identification macro only requires a dataset where each entry is attributed with a unique ID and the unique ID of the area downstream of itself. Use of the Downstream Area Identification macro is only one method for determining this downstream attribute. Other computerized methods, or even manual methods, are also suitable. Similarly, the Topologic Navigation macro only requires a dataset attributed with upstream and downstream element IDs. The method for the determination of these attributes is not relevant to the proper working of the macro. To test this assertion, the Upstream Area Identification and Topologic Navigation macros were modified to function on USGS Hydrologic Unit data (USGS-1, 2001). This data was specially attributed with the HUC ID of the area downstream of each area.

As discussed in section 5.4, the Topologic Navigation function determines the area downstream of a target area, and this downstream area then becomes the next target area. This process continues until the target area does not have any downstream areas (i.e it is an internal area, it is on the study area border, or it drains to the ocean). In the HUC dataset, areas draining to the ocean were manually assigned the code "-999" in order to match the ocean draining code produced by the Downstream Area Identification macro. This created compatibility between the Topologic Navigation macro and the HUC dataset.

The Upstream Area Identification macro was not significantly modified for use on HUC data. However, the data itself was significantly different. The HYDRO1K methodology for determining drainages is based on the assumption that each area may have one and only one downstream area, and that the upstream boundary of an area is usually the confluence point where two drainage areas meet. Within the HUC dataset, the drainage areas are defined administratively as well as by the landscape topology. As a result, more than two areas are upstream of most areas in the dataset. However, unlike in a Pfafstetter-attributed dataset, it is also possible to have only one upstream area,. In the HUC dataset, there exists at least one area with 7 upstream areas.



	J	K	L	M	N	O	P	Q
1	DWNSTR_HUC	Upstream1	Upstream2	Upstream3	Upstream4	Upstream5	Upstream6	Upstream7
77	17010212	17010206	17010210	17010207	17010209	17010211		
78	-555							
79	10050004							
80	-555							
81	10050012	10050002	10050007	10050008	10050010	10050011	10050005	10050009
82	10030203							
88	17010307	17010303						
89	17020016	17020013						
90	17030003							
91	17010213	17010208						

Figure 6.5 – Upstream Area Identification Results – HUC Data

Even with this increased number of upstream areas, the Topologic Navigation function still performs perfectly. The results of the navigation technique from HUC 10300102 are nearly identical to the results of the Pfafstetter-based navigation from area #8911. The main difference between the two navigation results is that the HUC dataset contains three internally-draining areas within North and South Dakota.

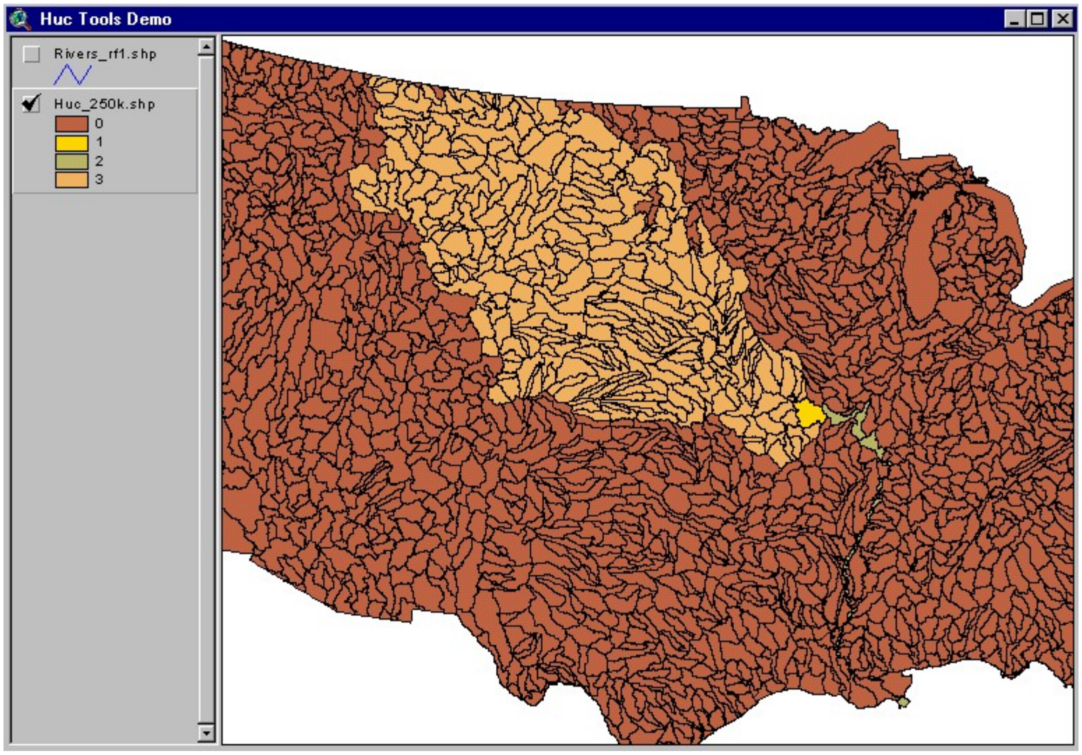


Figure 6.6 – Topologic Navigation from HUC #10300102

6.3 Navigation Capabilities in the ArcGIS Hydro Data Model

As shown in Section 6.2, all that is necessary to carry out the Topologic Navigation technique is a dataset attributed with area IDs and downstream area IDs. The Topologic Navigation techniques may be easily modified to conform with the format of any such dataset. For this reason, the **NextDownID** attribute was included in the “Drainage Systems” section of the ArcGIS Hydro Data Model (Maidment, 2001; Furnans and Olivera, 2001).

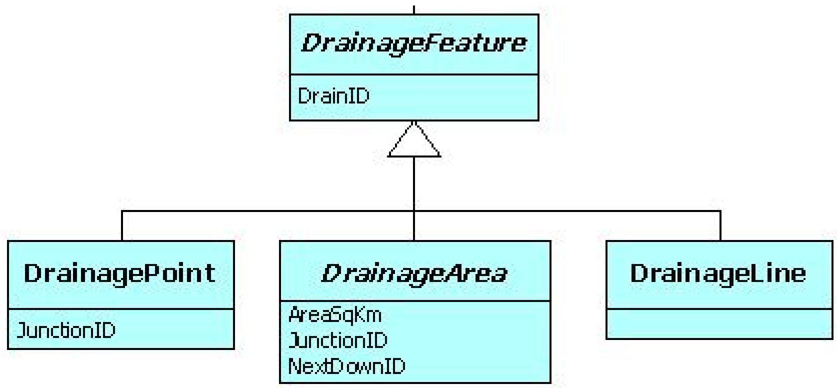


Figure 6.7 – UML Diagram for Drainage Systems – ArcGIS Hydro Data Model

The ArcGIS Hydro Data Model is a framework for storing geospatial data that has been developed by members of the GIS in Water Resources Consortium. This group is directed by Dr. David Maidment of the University of Texas at Austin Center for Research in Water Resources, and includes the Environmental Systems Research Institute (ESRI) and the U.S. EPA. The data framework is intended to be the standard format for storing, processing, and disseminating water-resources related geospatial data, allowing simplified interactions between future hydrodynamic, hydrologic, and water resources mathematical models.

Any hydrographic dataset formatted according to the ArcGIS Hydro Data Model will have a NextDownID on each drainage area within the database. Therefore, the Topologic Navigation technique is applicable to these datasets, for the Upstream Area Identification and Topologic Navigation macros have all of the input data they require. The Topologic Navigation technique is one of the first models/programs to be readily adapted to function in conjunction with the ArcGIS Hydro Data Model.

6.4 Issues with HYDRO1K – Pfafstetter code implementation

Throughout the development of the Pfafstetter Tools macros, many issues were raised concerning the USGS-Pfafstetter and Pfafstetter coding methodologies. Some of these issues have already been discussed in other sections, and are merely summarized in this section. The issues that have not yet been discussed are explained in detail. If applicable, a discussion is included for each issue in which the issue stems from an inconsistency between the Pfafstetter system developed by Otto Pfafstetter and by the USGS-Pfafstetter system incorporated into the HYDRO1K dataset.

Issue #1 – Drainage Density Problems

Within the USGS-Pfafstetter system, areas lacking 4 tributaries are not divisible into higher-level areas (Section 5.2.4). Such "drainage deficient" areas are assigned the highest-level digit "0," even though this digit is reserved for areas that are internal basins. This issue represents a sizeable discrepancy between the Pfafstetter system and the USGS-Pfafstetter system.

Issue #2 – Implied Topology for Internal Basins

Within the USGS-Pfafstetter system, lower level internal basins may be divided into smaller areas, which are in themselves also internal basins. These internal basins are assigned codes based on their relative geographic location, similar to assigning codes at the level 1 scale. However, through a simple analysis of the numerical internal basin codes, the internal areas are misconstrued as basins and interbasins that form part of a drainage system (Section 6.1). Therefore, the USGS-Pfafstetter methodology for assigning codes may imply incorrect topologies for internal basins. This issue represents a substantial drawback of both the Pfafstetter system and the USGS-Pfafstetter system. Neither of these systems is adequate for assigning codes to internal basins.

Issue #3 – Unknown Drainages – DEM remnants

Each HYDRO1K dataset contains a single area with the code “-1.” Such a code value is impossible according to the Pfafstetter system, and it is used in the USGS-Pfafstetter system in order to identify “artificial areas” that are created in the watershed delineation process from digital elevation models (DEMs). These “artificial areas” are often 1 DEM cell area in size and occur in areas of flat elevation where raster based watershed delineation is difficult. Such areas should be incorporated into the surrounding, properly coded areas and therefore eliminated from the database. The existence of these areas is directly proportional to the resolution of the DEM grid used to generate the watersheds. If the grid cells are large, then such areas are more likely.

Issue #4 – Incorrect Drainage Implications – Islands

The HYDRO1K dataset includes all landmasses on Earth, and is packaged on a continent-by-continent basis. Therefore, data for Japan is included with data for the Asian continent, and data for Madagascar is included with the data for the African continent. These islands, namely Japan and Madagasgar, have their own distinct drainage patterns, and are not in any way influenced by the drainage patterns on the mainland. Yet, within the HYDRO1K dataset, these islands are included as if they were part of the mainland drainage system.

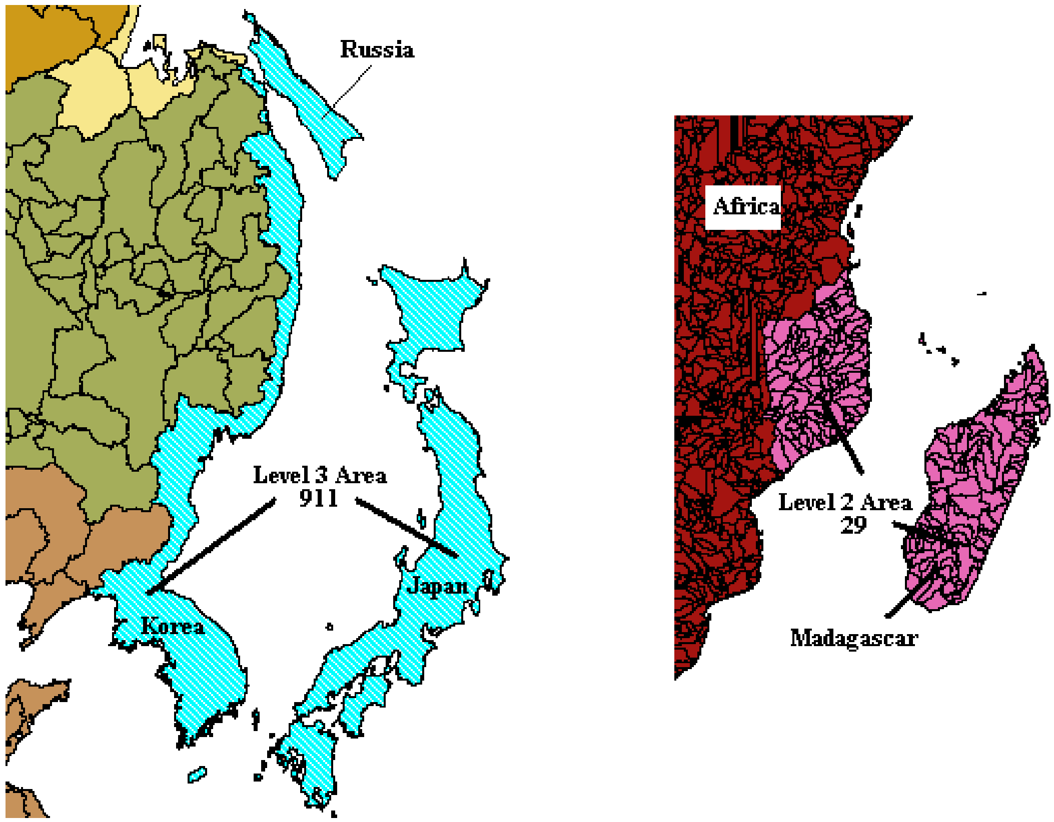


Figure 6.8 – Incorporating Islands into the Continental Codes – HYDRO1K data for Asia and Africa

This mis-representation of the global drainage patterns artificially increases the area (in square meters) of the drainage systems on the mainland, and it prevents the data user from manipulating only the data pertaining to the islands. This issue is due to the implementation of the USGS-Pfafstetter system, and is not at all existent in the Pfafstetter system.

Issue #5 – Representing Global Scale Data with Repeated Codes

The final evident issue concerning the USGS-Pfafstetter system and the HYDRO1K dataset is that the assigned codes are unique to each continental landmass, but not unique to the entire globe. For example, an area with the code #849 exists on each continent. Therefore in processing data from multiple continents, geographically distinct regions will have identical codes. Within the HYDRO1K dataset, numerical analyses of codes cannot determine to which continent a given area pertains. This issue is attributable to both the Pfafstetter system and the USGS-Pfafstetter system, although the documentation of the later mentions a solution to this problem (Verdin and Verdin, 1999).

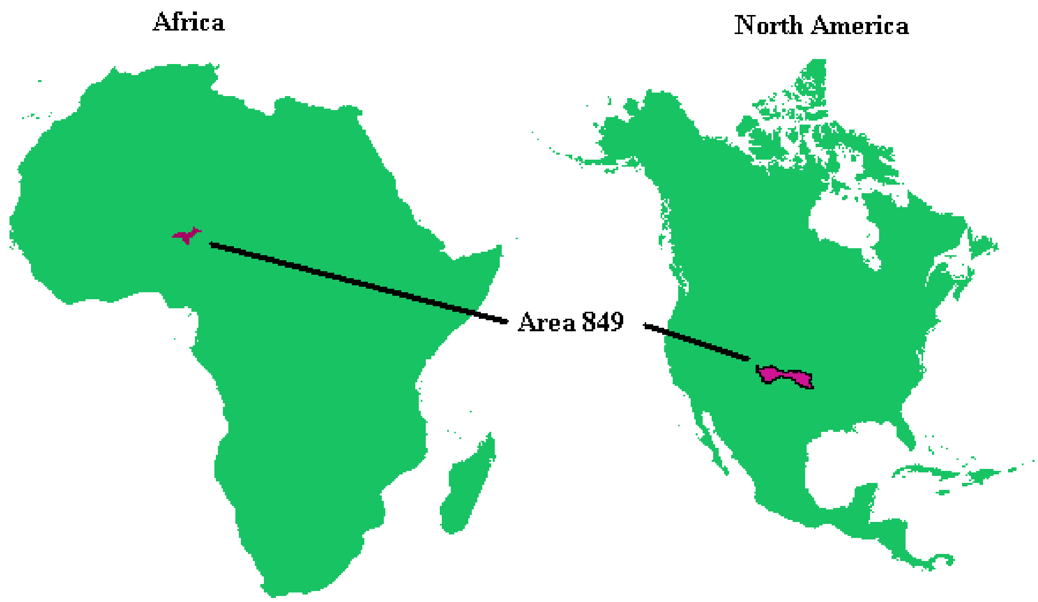


Figure 6.9 – Repeated Codes on the Global Scale

6.5 Suggested Alterations to Pfafstetter System

Each of the five issues discussed within the previous section are easily eliminated with simple modifications to the Pfafstetter coding methodology. These modifications, as well as some modifications that serve to clarify the coding procedure, are discussed in detail below.

Modification #1 – Divisions with up to four tributaries

The drainage density problems currently present in the HYDRO1K dataset all stem from the requirement of having four tributaries in an area in order to divide that area into a set of higher-level areas. This requirement assures that the divided area consists of at least 9 sections, with one section for each digit in the base-10 numbering system (Possibly excluding the internal basin and the "0" digit). With the Pfafstetter and USGS-Pfafstetter system, if an area does not contain 4 tributaries, it may not be further subdivided into higher Pfafstetter level areas. An area with only three tributaries may not be divided into smaller drainage areas, even though smaller drainage areas may be defined around the 3 tributaries and 3 main river segments. This "four tributary" requirement is convenient but unnecessary. A better approach is to divide areas into higher-level areas based on up to a maximum of four tributaries.

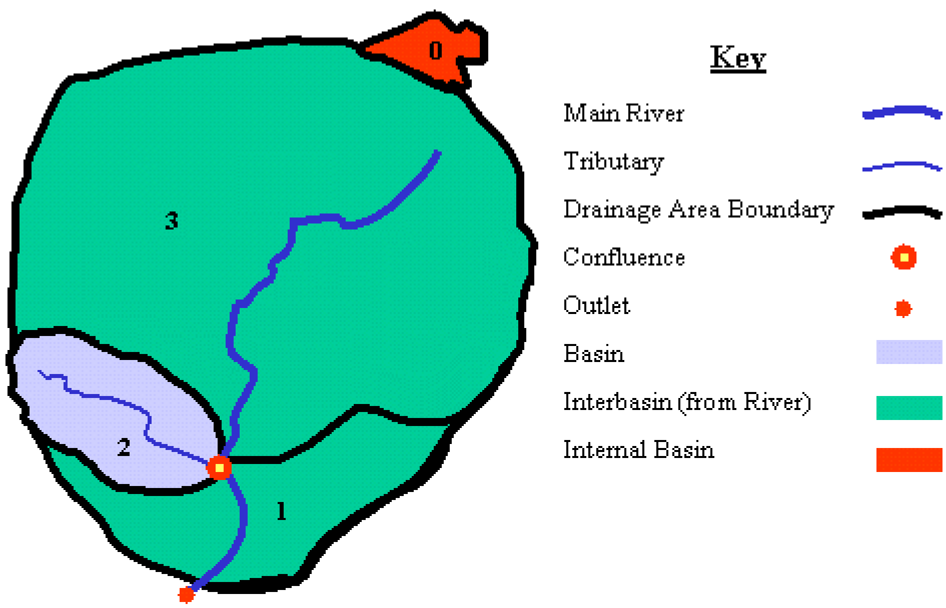


Figure 6.10 – Coding without the 4-Tributary Requirement

The Pfafstetter system is really based on sequential tributary intersections with a main river. In any non-coastal area at any level, the "1" area drains to the river between the outlet and the first tributary confluence. (For coastal areas, the coastline serves as the "river") The "2" area drains directly to this first tributary. If it were the only tributary, the rest of the overall area must drain to the main river, and it is given the "3" digit code (Figure 6-10). The procedure could easily be applied to areas containing 1, 2, 3, or 4 tributaries. The results would be areas with digits 1-3, 1-5, 1-7, and 1-9 respectively.

In each of these cases, the area with the largest digit number is always an interbasin, and the topology implied by the numbering system is still valid. Ignoring coastal areas, all areas with even highest-level digits will drain to areas with the code one less than their own. Area #2 drains to area #1, and area #6 still drains to area #5. Areas with odd highest-level codes will still drain to areas with odd highest-level codes two less than their own, with the exception of areas with the highest level digit "1." These areas will drain to other areas as predicted with the **identify** subroutine (Section 5.2.4). In fact, the Downstream Area Identification results will not be affected by this modification, because the downstream relationships between codes are unchanged. The use of "search ranges" in determining the area downstream of interbasins with the highest-level digit "1" assures that the determinations are made correctly.

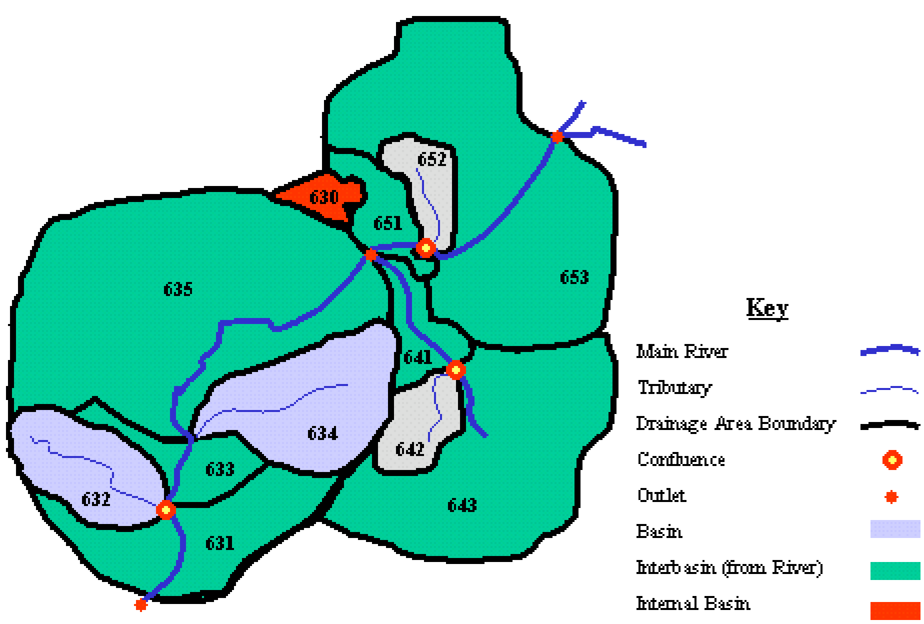


Figure 6-11 – Downstream Relations without the 4-Tributary Requirement

As determined in the identify subroutine the search range for the level 3 interbasin #651 includes areas 630-639. Therefore, the downstream area will have the greatest code within this range. This code could be 630, 631, 633, 635, 637, or 639 assuming the identify subroutine is applied to a complete Pfafstetter-attributed dataset. Codes 632, 634, 636, and 638 correspond to basins, which may not receive upstream flows. If areas with these numbers exist, then the interbasin with the code one greater also exists. Therefore these codes will never be selected out of the search range. The code of the downstream area, therefore, is dependent upon the number of tributaries that were used in creating the downstream areas from the next lowest level area. If three tributaries exist in the level 2 area #63, then the area downstream of area #651 is area #637. If only two tributaries exist, then the downstream area is #635 (Figure 6-11).

A corollary to this modification arises in situations where an area that is to be divided into higher Pfafstetter level areas does not contain a single tributary. Within the USGS-Pfafstetter system, such an area would be assigned the "0" highest level digit because it is drainage deficient. With this modification, this area would be assigned the highest-level digit "1."

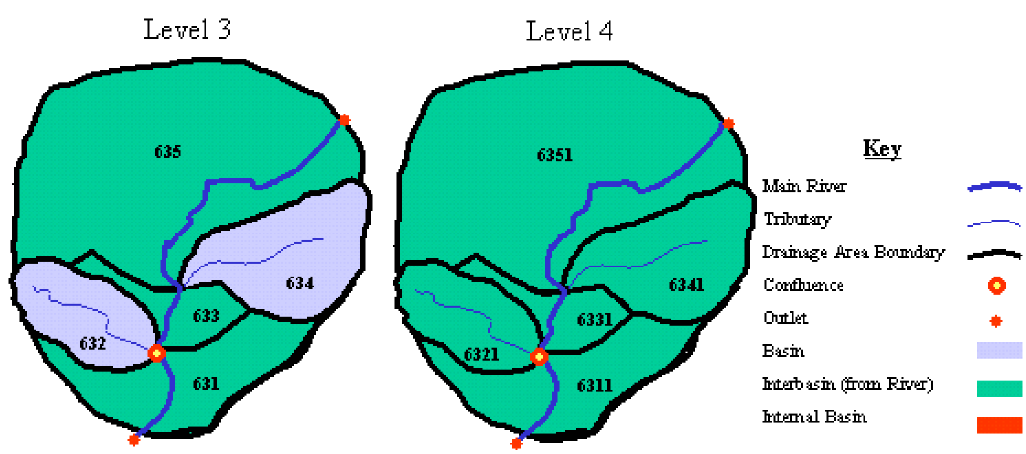


Figure 6-12 Areas without tributaries are assigned the "1" digit at higher levels (See Modification #2 for discussion of interbasin-basin definitions)

By assigning the “1” digit to areas without tributaries, the downstream relations are maintained and may be determined with only the identify subroutine. The drainage density problem within the USGS-Pfafstetter system is thereby eliminated. Only internal basins would be assigned the “0” digit, and the confusion associated with the “0” digit within the HYDRO1K dataset would be removed. However, some areas that are basins will receive the “1” highest-level digit, which is traditionally reserved for interbasins (Figure 6.12). For example, if the level 3 area #632 does not contain a single tributary, then this same area receives the level 4 code 6321. This code suggests the area is an interbasin. It would be incorrect to assign such areas the “2” digit, because this would incorrectly imply that the area #6322 drains to area #6321, when area #6321 does not exist. The only choice is to assign the area the “1” digit and to modify the definitions of interbasins and basins so as to account for this new numbering scenario.

Modification #2 – Adjusting the Area Definitions

As given in Table 4.2 and Verdin and Verdin (1999), a Pfafstetter basin is defined as the area drained by a tributary. A Pfafstetter interbasin is the area draining to a reach of the main river between two tributaries. Basins are assigned even digits, and interbasins are assigned odd digits (Verdin and Verdin, 1999). While simple to understand, these definitions and numbering characteristics are not applicable to all drainage areas within the HYDRO1K dataset, even if modification #1 is implemented. An alternative, more encompassing set of definitions is given in Table 6.1.

Table 6.1 Alternative Definitions and Rule for Pfafstetter Drainage Areas

Basins:	The area draining the entire length of a river that flows to the drainage area outlet
Interbasins:	The area draining to a segment of a main river or coastline
Internal Basins:	Areas that do not contribute flow to other drainage areas or to the ocean
Application Rule	If an area meets both of the new definitions for a basin and an interbasin, then the definition of the interbasin prevails when dictated by the Pfafstetter level.

Under these definitions and application rule, all basins still have even highest-level digits, and all interbasins still have odd highest-level digits. The new definitions incorporate the definitions from Table 4.2 and at the same time expand the meaning of the terms "basin" and "interbasin." The definition of an internal basin is unchanged. To understand the reasoning behind these definition modifications, it is necessary to consider each drainage area type with respect to the coastline.

The original Pfafstetter definitions of basins and interbasins fail when considering continental-scale drainage areas and coastal zones. At the continental scale, the basins drain to rivers flowing directly into the ocean. It would be incorrect to call these rivers "tributaries" as is required by the old definition of a basin. Doing so would suggest that the rivers contribute flow to some other, larger river, which is not the case. These rivers do flow into a linear feature, namely the coastline. However this linear feature is not analogous to a river as it does not carry overland flow along its length. By defining a basin as an area that drains the entire length of a river, this level 1 contradiction in terms is avoided. It is accurate to say that basin #8 on the North American continent contains the entire length of the Mississippi River system. Therefore area #8 fits the new definition of a basin.

The inclusion of a coastline also disrupts the old definition of an interbasin, because coastal interbasins do not drain to a section of the main river between two tributaries. They drain to a section of the coastline bounded by the points of intersection between the coastline and rivers draining coastal basins. For this reason, it is necessary to incorporate the term "coastline" in the definition of the interbasin.

With these new definitions, it is possible for a single area to meet requirements for being a basin and an interbasin. One such situation involves the "X9" areas where "X" is an even number (Figure 6-13).

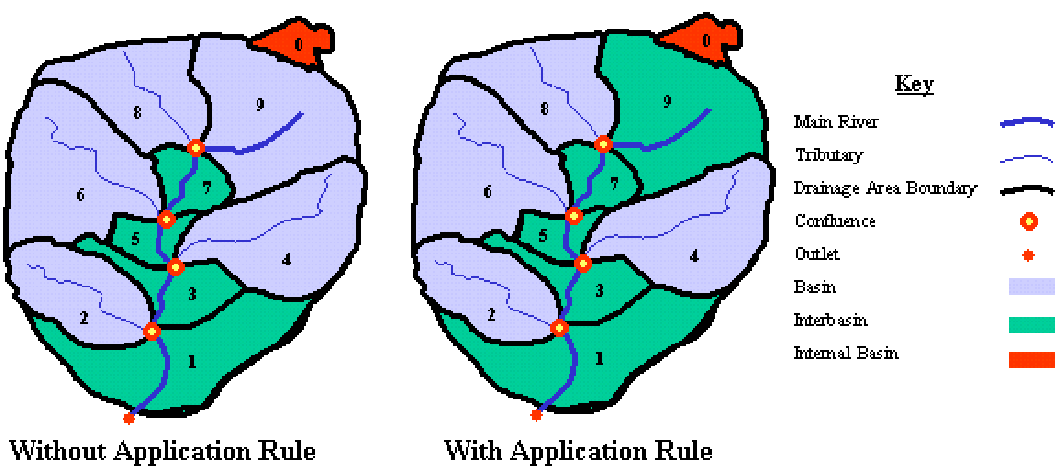


Figure 6-13 – Altered Classification of "9" based on the Application Rule

When any basin is subdivided into a higher Pfafstetter level, the two most upstream higher-level areas will each meet the new definition of a basin. One area will drain to a tributary, and the other area will drain to the most upstream section of the main river. This area meets the definition of a basin because it contains the entire length of the river that drains to its area outlet. However, this area also meets the definition of an interbasin because the area drains to a section of the main river at the previous Pfafstetter level. For example, level 1 basin #6 is divided into level 2 areas #61-#69. Area #69 is the area most upstream within the level 1 area #6, and it drains to the entire length of the river it contains. Therefore, it is a basin. However, at the level 1 scale, this river forms part of the main river, and therefore the area drains a main river segment. Therefore, it meets the definition of an interbasin. With the application rule, the area is classified as an interbasin, and it may therefore receive an odd numbered digit (Figure 6.13).

These new definitions and the new application rule will also apply if the “4-tributary” requirement is lifted in favor of modification #1. As previously discussed, this would produce situations in which areas without any tributaries would be given the “1” digit at higher Pfafstetter levels. Under this system, basins would receive an odd digit at each subsequent higher Pfafstetter level, which is in contradiction with the normal Pfafstetter pattern (See area #6341 in Figure 6-12). However, at the higher level scales, the river contained within this drainage area is the main river for the area, and therefore with the application rule the area is properly numbered as an interbasin.

These new definitions and application rule describe drainage areas with certain distinguishable characteristics depending on the analysis scale. This is a small, yet important, difference from the definitions employed by Verdin and Verdin (1999), and these differences are not important when Pfafstetter-based numbering schemes are used for local scale, inland area applications. For example, at a global or continental scale, the coastline term becomes important in the definition of an interbasin. This term is not relevant for the high-resolution drainage areas at continental interiors. Coastline features do not exist for the local drainage areas in Kansas, and as such all of Kansas’ interbasins drain to main river segments. At such a local scale, the definitions used by Verdin and Verdin (1999) are perhaps more intuitive and easier to understand. However, for the global scales and geographic extent of the HYDRO1K dataset, the modified definitions are necessary.

Modification #3 – Adjusting the Numbering of Internal Basins

The Pfafstetter system is not easily adaptable for use on internal basins because such drainage areas often do not have topographical relations with the drainage areas around them. This difficulty is apparent within the HYDRO1K dataset, where the assignment of IDs to internal basins incorrectly implied non-existent drainage relationships (Section 6.1). As discussed in Section 5.2.4, it is theoretically possible to assign Pfafstetter-based codes to certain internal drainage areas that exhibit distinct drainage relationships. However, these drainage situations are extremely ideal and rare in nature. Most internally draining systems are mis-represented by the Pfafstetter and USGS-Pfafstetter systems.

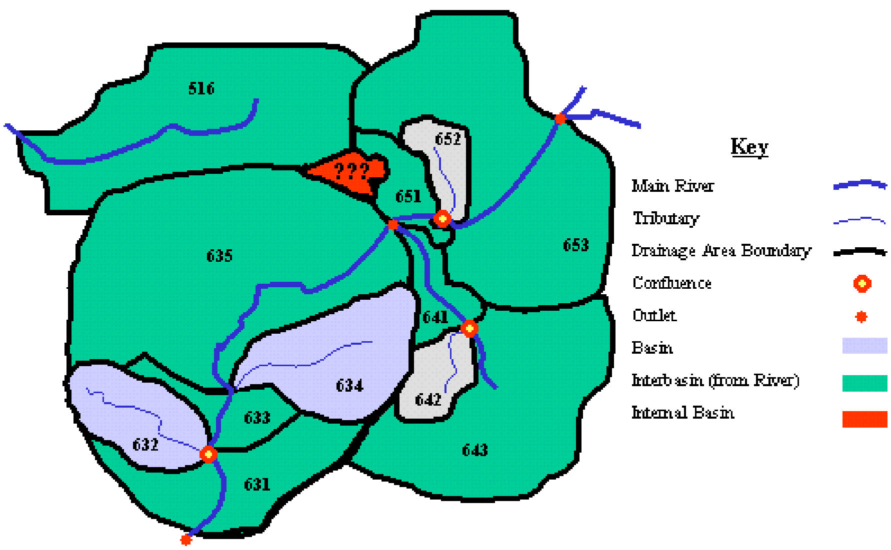


Figure 6.14 – Assigning Codes to Internal Basins – to which area does the internal basin belong?

According to Verdin and Verdin (1999), internal basins are given the IDs of the lower level surrounding drainage area with the “0” digit as the highest-level digit. This methodology works well if the internal drainage area is incorporated within an interbasin or a basin. However, if the internal basin is not within such an area, then its numbering may be problematic. The internal basin in Figure 6-14 may take on the level 3 codes 510, 630, or 650 because it is bordered by the level 2 areas #51, #63, and #65. Each of the three codes is equally valid, and the decision is likely made based on information of other internal basins within the area. For example, if other internal basins exist that may be attributed to areas #63 and #65, then the internal basin in Figure 6-14 is assigned the level 3 code 510. This decision-making process involves an analysis of the geographic location of different areas, and is therefore not following the topologic principles guiding the Pfafstetter system.

Another difficulty with the Pfafstetter system and USGS-Pfafstetter system treatment of internal basins is that only one internal basin may exist at each level, regardless of how many internal basins actually exist at this level. If multiple internal basins exist, then the largest basin is assigned the “0” digit and is recognized as an internal basin. Any other internal basin is simply incorporated into a surrounding drainage area, thereby incorrectly inflating the extent of that basin or interbasin. In this way, the Pfafstetter and USGS-Pfafstetter systems incorrectly represent the drainage patterns of areas with internal basins.

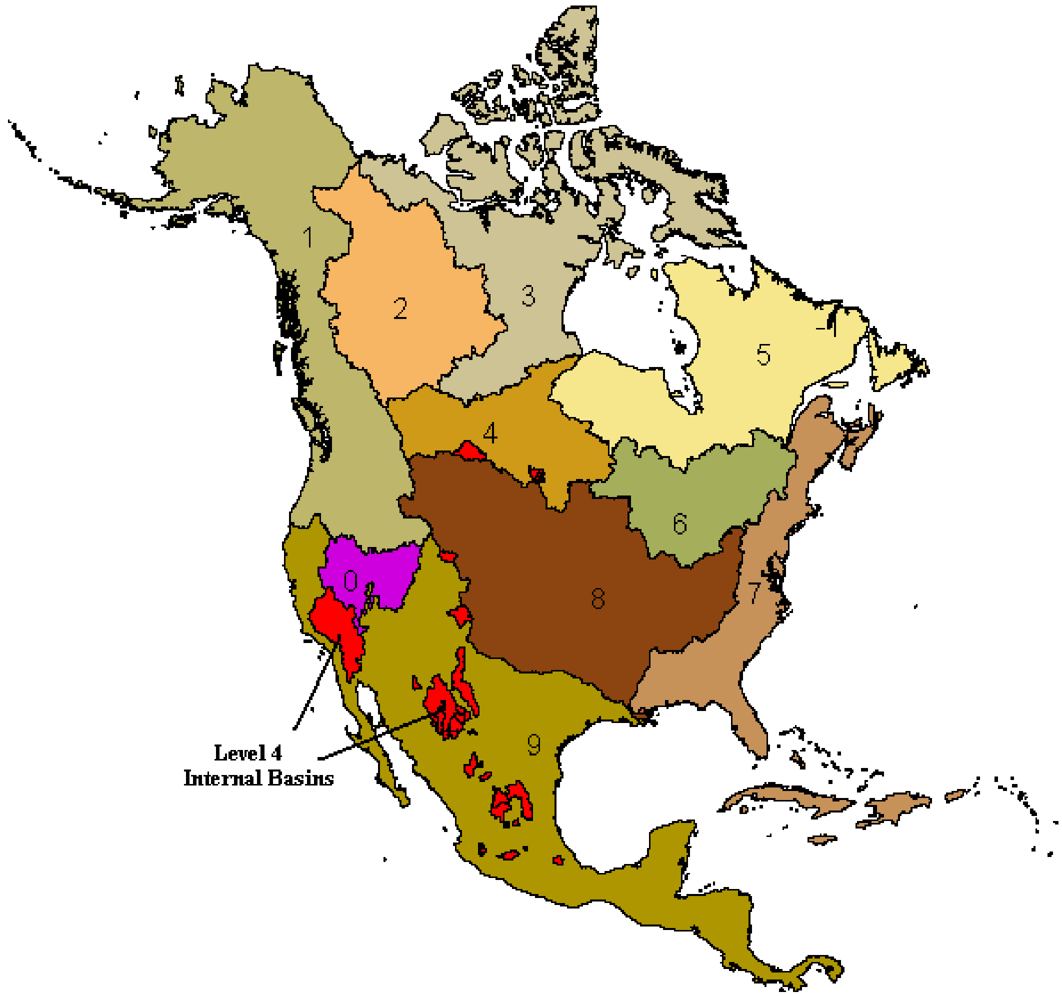


Figure 6.15 – Level 4 internal basins are not distinguished at the level 1 scale (HYDRO1K Dataset for North America)

Examples of this inaccuracy are evident within the HYDRO1K dataset for North America. The internal basin at the level 1 scale for this dataset is the basin containing the Great Salt Lake in Utah. No other internal basins exist at the level 1 scale because they were incorporated into their surrounding interbasins and basins. However, numerous internal basins exist at the level 4 scale, and these areas are distinct from the level 1 "0" internal basin. These level 4 internal basins are always internal basins, regardless of scale, in that they do not contribute flow to other areas. At the level 4 scale, the drainage characteristics of these internal areas are represented, yet at the level 1 scale they are ignored (Figure 6-15).

Given these difficulties, the best method for numbering internal basins is to assign them codes that will obviously distinguish them from interbasins and basins at all scales and at all levels. Internal basins should be assigned unique negative codes at the level 1 scale. The values of these codes would not reflect any type of geographic position relative to non-internal basins, and they should be numbered according to size. Therefore, the largest internal basin is assigned the code “-1,” the next largest internal basin is assigned the code “-2,” etc. until all of the internal basins in the dataset have been numbered. The only exception is that numbers divisible by 10 are not used in identifying internal basins. Therefore, the 10th largest interbasin gets the code “-11.” The “0” digit is again reserved for a special meaning as described later. The numbering of internal basins in this way also ignores the idea of Pfafstetter levels by allowing any number of digits to identify drainage areas at equal Pfafstetter levels. This is allowed because with this modification, internal basins are removed from all of the constraints of the Pfafstetter system. The key, therefore, in distinguishing internal basins (which do not follow the regular Pfafstetter numbering scheme) from those areas that do follow the numbering scheme is the use of the negative sign. This sign, because it is a numerical operator, is easily recognizable by programs such as Microsoft Excel, and therefore it would not hamper any numerically based programs that work with Pfafstetter codes. In fact, the Pfafstetter Tools, as they exist for use on the HYDRO1K dataset, would simply ignore all of the areas with negative codes and would still carry out their various functions.

The topographic relationships employed by Pfafstetter-based numbering systems are still useful for internal basins if the internal drainage areas receive flow from other basins and interbasins. According to the definition of an internal basin, such an area does not drain to another area, yet it may still receive flow from other areas. These areas will also not contribute flow to any river system that flows into the ocean, and their assigned ID codes must reflect this characteristic.

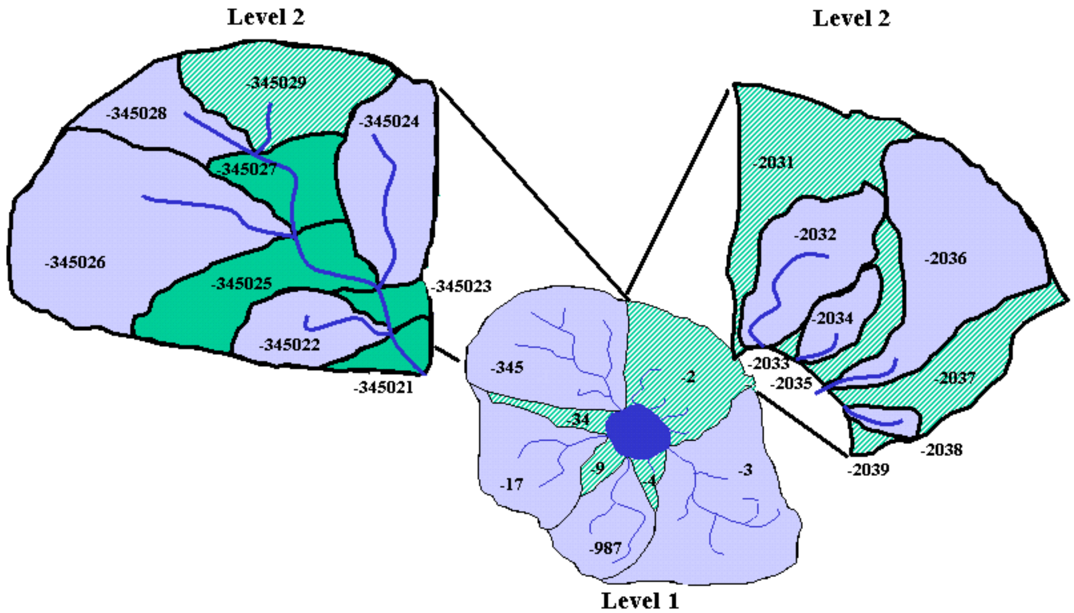


Figure 6.16 – Pfafstetter-based numbering of drainages to internal basins (Compare with Figure 5.24)

In numbering areas that form part of drainage systems that drain into internal basins, the standard Pfafstetter numbering techniques may be used. However, in doing so, it becomes important to re-subject the internal basins to the concept of Pfafstetter levels, after all of the internal basins have been given unique negative numbers. This is accomplished with the "0" digit, which is appended onto the internal basin's unique identifier only if that basin may be divided into higher-level areas. As shown in Figure 6.16, area #-345 is an internal basin at the Level 1 scale. However, this area may be divided into higher-level Pfafstetter areas. These level 2 areas are numbered with the "-345" followed by the "0" digit. The next digit "2" signifies that the area to be subdivided has characteristics of a basin and that it must be subdivided as a non-coastal area. The next and final digit is the appropriate digit as assigned within the Pfafstetter and USGS-Pfafstetter systems. A similar process is employed to assign the level codes to the level 1 interbasin #-2. These new areas each have the code “-2” followed by a “0” and a “3,” where the “3’ signifies that the area is part of an interbasin. The remaining digit is assigned based on the methodology for numbering coastal areas.

In this example, the level 2 codes consist of 6 and 4 digits, respectively. This is contrary to the Pfafstetter system trait that the number of digits equals the number of hierarchal levels for which the area belongs. However, the appropriate level for an internally – draining area (i.e. and area draining to an internal basin) may be determined by counting the number of digits to the right of the “0” digit.

The drawback of this method for numbering internal basins is that it is less intuitive and more difficult for a data user to understand. However, a computer program could easily be developed to recognize relationships between such areas. Also, because the areas are not numbered relative to a drainage system, the geographical proximity of areas may not be estimated solely based on an analysis of the assigned codes. This may or may not be a problem, depending on the purpose for which the data is used. The benefits of this methodology are that internal basins are recognized at every scale and level, and that their surrounding drainage systems will have accurate area and flow attributes (e.g. flow can be calculated based on area, precipitation data, and land cover, for example). Also, by combining the numbering scheme with modification #1, the density subroutine within the Downstream Area Identification macro becomes entirely unnecessary. Therefore the Downstream Area Identification process would be faster and more efficient. This macro could also easily be modified to recognize situations in which areas drain to internal basins, so that downstream areas could be determined. The errors in downstream determinations for HYDRO1K internal areas would be eliminated (See Section 6.1).

Modification #4 – Assigning unique codes for each world landmass

Although suggested in Verdin and Verdin (1999), the Pfafstetter and USGS-Pfafstetter systems (as employed in the HYDRO1K dataset) do not provide methods for distinguishing global locations based on the numerical ID. Each continent has areas with codes identical to those on other continents. To fix this problem, an arbitrarily defined set of "landmass" codes needs to be employed. These codes would serve to uniquely identify every landmass on Earth, including the continents and islands such as Japan and Madagascar. (See Issues #4 and #5 in Section 6.4). These codes would be numerical codes of as many digits as necessary to uniquely identify each separate landmass. The codes and their respective landmasses could be stored in a simple computational look-up table that is produced with the dataset of interest or standardized for use all over the world.

The proposed landmass codes would be the first few digits of the area ID number, and they would be separated from the rest of the ID number by the "0" digit. For example, if the code for the North American landmass were "23" then the code for the Mississippi River drainage system would be "2308." The use of the "0" digit as the separator between the landmass code and the Pfafstetter-based code would not cause any confusion with internal basins or drainage deficient areas as long as modifications #1 through #3 are also employed in the numbering system. The only potential source of confusion would be in coding basins that drain internal basins, because these areas already have a "0" digit separator. In such a situation, the first "0" separator would indicate the separation between the landmass code and the internal area code, and the second "0" digit would indicate the separation between the internal area code and the Pfafstetter-based code. For example, an area with the code "-470348023" would be an area draining to internal basin "348" located on landmass "47." It would be a level 2 interbasin within a level 1 basin.

One problem with this numbering system is that it further eliminates the "economy of digits" characteristic of the Pfafstetter and USGS-Pfafstetter systems.

One way to avoid this would be to store the landmass code as a second attribute of each drainage area in the database.

Modification #5 – Continental Scale Considerations

The current Pfafstetter and USGS-Pfafstetter systems rely on relative geographic locations of areas in order to assign level 1 codes. At the level 1 scale, the area with the “2” code is the basin whose outlet is the most northern of all the basin outlets. This is an arbitrary decision, but it is functional. However, this method does not allow assignment of ID codes without resorting to geographical references. A better method would be to assign the “2” code to the level 1 basin with the largest drainage area.

Such a basin would be easily identifiable based on a calculation of the upstream area extents of all areas within the dataset. This calculation is required in assigning Pfafstetter-based codes, and therefore does not require significant new work. The remaining basins could be identified by navigating along the coastline from the “2” basin outlet until the outlet from an area with the 2^{nd}, 3^{rd}, or 4^{th} greatest upstream areas is located. This follows the concept of level 1 code assignment within the USGS-Pfafstetter system, and does not consider geographic location. Then next basin along the coastline from the “2” basin gets the “4” level 1 code, and the area draining to the coastline between these two outlets would get the “3” level 1 code. This process would continue until all of the basins and interbasins are identified and numbered.

The current USGS-Pfafstetter system also uses the geographic location of areas in deciding the boundary between the level 1 and level 9 interbasins. This boundary may be a notable geographic feature, or it may be the border of a local drainage area (Verdin and Verdin, 1999). The decision is arbitrary. An alternative method is to define the border in order to equalize the total areas of interbasins “1” and “9” to the greatest extent possible. This equalization would involve a simple optimization routine, which could be implemented in the code-assignment program.

The benefit of these alterations in continental-scale code assignment is the elimination of references to geographic locations. These solutions are conventions that can be substituted for the conventions employed in the USGS-Pfafstetter system. The drawback of this modification is that some geospatial information is lost with regard to the relative positioning of the level 1 basins. However, this is not that important because the “2” area does not necessarily drain to the landmass’s most northern point (See Figure 6.15). Also, area “3” is still always located between area “2” and area “4.” However in traveling from area “2” to area “4” the direction along the coastline is not specified, but can be determined by the program performing the code assignment. The data user would have to understand this slight limitation in methodology.

Modification #6 – Allowing for local changes

Once a dataset is attributed with Pfafstetter codes, additions to the drainage areas may be difficult. It would be relatively easy to merge existing drainage areas together, and therefore take the code of the most downstream area as the code of the new merged area. This would remove at least one code from the dataset, which could cause problems in running the Downstream Area Identification macro. In such a case, the Downstream Area Identification macro would determine that that the areas immediately upstream of the merged areas were on the study area border. The user would then need to manually adjust the downstream area codes for the areas in question.

A greater challenge is involved in the addition of drainage areas to the dataset. For example, a data user who is interested in water availability modeling may need to determine the areas upstream of a given control point. This point may or may not coincide with an outlet of an area with a Pfafstetter-based ID. For example, if the user’s water withdrawal point is located along the main river in interbasin #6343, then all areas upstream of interbasin #6343 drain to the point. The user can determine the part of area #6343 that drains to the withdrawal point (via raster based delineation methods or cartography), which is a “partial” area. This partial area combined with the upstream area identified with the Topologic Navigation macro would be the area upstream of the point. It is likely that the user would want to preserve the partial area within the dataset, and for this the area would need some type of Pfafstetter-consistent ID. This ID should be such so that the Topographic Navigation technique still produces accurate results on such a modified dataset. The Pfafstetter and USGS-Pfafstetter systems do not account for such a condition. A solution to this problem is to use decimal values.

A decimal value seems most appropriate because the new area is a part of a previous area within the database (Figure 6.17). The new code should be equal to the code of the area from which it was derived, plus the lesser of ½ of the absolute difference between the original area’s code and the next higher code in the database or 0.5. For example, if a new area is made within area 433, the new area’s code is:

$$Code = 433 + \frac{434 - 433}{2} = 433.5$$

If this new area if then split into two other areas, the new area’s code becomes:

$$Code = 433.5 + \frac{434 - 433.5}{2} = 433.75$$

In this way, any area can be divided indefinitely (in a mathematical sense, at least), and a unique numerical code will always be assigned. Also, the downstream relationships between these newly created areas are easily determined. If an area is found to have a code with a non-zero decimal component, then its downstream area is the area with the largest code still less than its own. A simple modification to the Downstream Area Identification macro is all that is needed to produce these results. The Upstream Area Identification and the Topologic Navigation macros would not need to be modified.

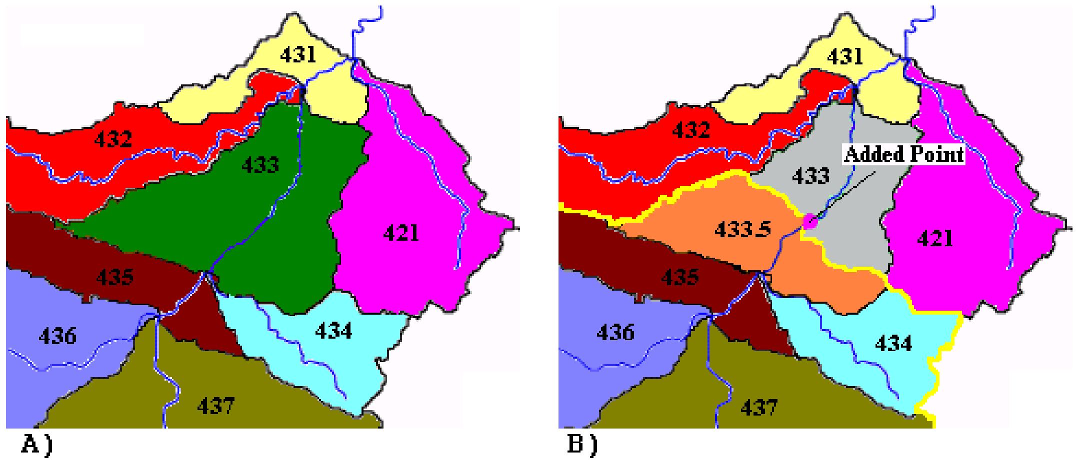


Figure 6-17 – Numbering additional areas – A) Original area #433, B) New area #433.5 is upstream of the new area #433. The area upstream of the added point is outlined.

Chapter 7 – Conclusions and Future Work

This thesis explores the spatial structure of drainage area connectivity. In particular, it discusses methods for tracing water movement from one drainage area to the next through the landscape, and for using this connectivity to derive the entire upstream and downstream drainage areas of a single area in the landscape. Existing methods for describing connectivity each require either raster data or geometric networks. The purpose of this effort was to develop a new method that would function on polygon datasets. The method was to be applicable to a wide range of datasets, and was to be executable with or without ArcGIS software. This new method is called Topologic Navigation.

Topologic Navigation is the process of determining drainage area relationships amongst entries in a database. Each area must have as attributes a unique ID code, as well as the code of the area immediately downstream of itself. Based on an analysis of the downstream ID codes, it is possible to determine the areas immediately upstream of each area in the database. This information, in turn, may be used to determine all of the areas upstream and downstream of any selected area in the database.

Downstream area codes are discernible for datasets attributed according to Pfafstetter-based coding systems. These numerical systems assign ID codes to drainage areas based on the landscape topology. Through an analysis of the specific code assigned to each drainage area, it is possible to determine the code of the area immediately downstream. This determination is unequivocal on properly coded datasets, and it is made without reference to any geographic information source (such as river networks, maps, or GIS systems). Pfafstetter-based numbering systems are the only systems from which topology can be discerned without the additional use of geographical references.

In this work it has been demonstrated that Topologic Navigation is a viable methodology for determining upstream and downstream relationships between drainage areas. Upstream and downstream navigations have been successfully performed on datasets attributed according to Pfafstetter-based systems, as well as on the HUC dataset for the United States. This demonstrated the applicability of the navigation technique to datasets from various sources. The success on two different datasets with different formats implies the technique is also theoretically applicable to any dataset with connectivity between entries.

Although the Topologic Navigation methodology is only discussed with regard to drainage areas in this work, it is equally applicable to other drainage elements and river networks. The Topologic Navigation macro will function correctly as long as the dataset to which it is applied contains the requisite downstream ID attributes for each database entry. The other features represented in the dataset do not affect the proper functioning of the macro. This is the first GIS-related navigation technique that is equally applicable to polygon, arc, and point datasets. The technique has this ability because it considers only the attributes within the database and not the geospatial characteristics of these attributes.

The Topologic Navigation technique is unique in that it is functionally independent of the network analysis tools and raster extensions in the ArcGIS system. The technique could be developed into a stand-alone program, or it could be incorporated into the functionality of any COM-compliant database software (including but not limited to ArcGIS and Microsoft Excel). Compatibility with Microsoft Excel is especially promising, for it allows potential river basin modelers and planners to implement established Microsoft Excel hydrologic applications without the need for modifying these applications for use in the ArcGIS environment.

Differences have been demonstrated between the Pfafstetter and USGS-Pfafstetter systems for numbering drainage areas. Despite these differences, both systems are largely successful at representing unambiguous topological relationships among most drainage areas. However, the systems do need to be modified and standardized so as to better describe the topology of all drainage areas, particularly the internal basins.

Many conclusions may be drawn from this work, specifically with respect to the assignment of drainage area IDs in a topologically descriptive manner. It is difficult to develop a code assignment methodology that is 1) based on drainage patterns, 2) equally applicable to all datasets, 3) useful in predicting drainage-relationships between areas, and 4) easy to understand. Also, it is difficult to develop a methodology that produces ID codes from which topologic relationships between areas are completely defined without reference to geographical information sources. Any numbering scheme that exhibits all of these characteristics could be considered an "ideal" scheme for navigation purposes. Each of the numbering schemes presented in Chapter 2 (i.e. the ORSTOM method, the NWIS method, the USGS HUC system, etc) have some, but not all, of these characteristics. The Pfafstetter and USGS-Pfafstetter systems have more of these characteristics, but do not quite match the "ideal" numbering system.

The Pfafstetter numbering system is a methodology useful on only the most basic and consistent of drainage area datasets. As it is based on drainage patterns, it may be used to predict drainage-relationships and is easy to understand. However, without modification it is not applicable to all datasets. The system involves the topology implied by river networks that must fit a standard pattern where only two rivers meet at a confluence. Situations where three or four rivers converge at a single confluence are not addressed within the system. Also, each drainage area must contain at least four tributaries to the main river in order to be subdivided. The system does not consider the possibility that less than four tributaries may exist within the drainage area, and it does not include a coding methodology for handling such a situation. However, datasets exist that contain three-river confluences and areas with less than four tributaries. The Pfafstetter system may not be used to assign codes to such datasets, unless the system is altered.

The USGS-Pfafstetter system, as discussed in this work, is a revised version of the Pfafstetter system. The USGS system makes use of the basic Pfafstetter numbering theory in the way codes are assigned to coastal and non-coastal areas. The system also incorporates alterations to the theory in order to assign codes to areas that do not meet the standard characteristics required by the Pfafstetter system. The alterations make the USGS-Pfafstetter system applicable to describe the topology of elements within global datasets such as the HYDROK1 dataset. However, as discussed in Chapter 5 and Chapter 6, this system contains instances where assigned codes incorrectly imply drainage connectivity between areas that are topologically distinct. In most cases, the incorrectly implied drainage relationships involve areas that have codes assigned based on geographic locations. For example, Section 6.1 describes errors in the Downstream Area Determination results for the internal basins in the vicinity of the Great Salt Lake in the Western United States. These errors are avoided if the modifications listed in Chapter 6 are incorporated into a new “CRWR-Pfafstetter” numbering system. This new CRWR-Pfafstetter system would completely eliminate all geographic references from the code assignment process, making it the most “ideal” of all of the systems developed so far. However, upon incorporating the modifications listed in Chapter 6, the CRWR-Pfafstetter system would be more complex than the original Pfafstetter or even the USGS-Pfafstetter systems.

The USGS-Pfafstetter system was devised in order to more accurately apply the Pfafstetter numbering theories to actual drainage area datasets. The Pfafstetter system evolved into the USGS-Pfafstetter system so that codes could be assigned to the HYDRO1K dataset. The modifications presented in Chapter 6 of this work represent the next possible evolutionary step for the Pfafstetter-based family of numbering systems. The evolution analogy is appropriate because each successive numbering system is an adaptation of the older systems, and each adaptation is more efficient at describing the topology of the dataset to which it is applied. Consistent with biological evolutionary processes, successive adaptations of the basic Pfafstetter theory yield numbering systems of greater and greater complexity, with attendant loss of clarity. This loss of clarity makes systems less and less "ideal" because they are more difficult to understand. However, a more complex Pfafstetter-based system would be preferable as long as it completely describes the landscape topology. To do this, it must completely eliminate all geographical referencing from the code assignment process.

A CRWR-Pfafstetter system that includes the modifications discussed in Chapter 6 would accurately describe the landscape topology without reference to geography. Unlike in the other Pfafstetter-based systems, level 1 areas would be assigned codes based on basin area size and relative location along the coastline. Internal basins would not be assigned codes based on their proximity to and northern location relative to other internal basins. The Downstream Area Identification process would produce error free results, and the problems associated with the internal basins in the USGS-Pfafstetter system would be avoided. The Topologic Navigation process could be applied to error-free data, thereby producing accurate results every time and for every drainage area.

The CRWR-Pfafstetter system would be the only Pfafstetter-based system to incorporate the fact that internal basins do not interact with the drainage areas around them. Internal basins do not drain to other basins, and at the lower levels they will likely not receive flow from other drainage areas. They are therefore distinct units that should not be associated with their surrounding drainage areas. The internal basin numbering methodology described within Modification #3 of Section 6.5 is the only methodology that completely distinguishes between internal basins and drainage areas from which runoff eventually reaches the ocean. Although the CRWR-Pfafstetter numbering scheme for internal basins is complex and reduces the “economy of digits” (Verdin and Verdin, 1999) characteristic of the other Pfafstetter-based systems, each internal basin code would be easily recognizable both by a computer and by the data user. This is not the case for internal basins in the USGS-Pfafstetter system, which must be distinguished from drainage deficient areas. The benefits of this more complex system greatly outweigh any inconvenience the complexity of the system imposes on the user.

The Pfafstetter Tools macros, as discussed in this work, perform the Topologic Navigation technique on data with or without Pfafstetter-based attributes. Despite these successful applications, the macros are not yet perfected. The entire Pfafstetter-based Topologic Navigation technique may be improved and enhanced through various methods described in the following sections.

7.1 Completion of the Pfafstetter Tools for Navigation Support

One drawback of the Pfafstetter-based Topologic Navigation technique is the relative dearth of existing datasets that are attributed with Pfafstetter-based codes. Once completed, the Node Swap, Stream Filter, and Code Assignment macros discussed in Section 5.5 will extend the applicability of the Pfafstetter-based Topologic Navigation technique to many more existing and future drainage area datasets. The on-going development of the five Pfafstetter Tools macros discussed in Section 5.5 will be described in detail in the companion report to this thesis (Furnans and Olivera, 2002).

7.2 Implementation within the ArcGIS System

Once the Code Assignment function is fully operational, the next step is to incorporate the Pfafstetter Tools into macros that may be accessed directly within the ArcGIS software. The Pfafstetter Tools could work as macros within the

ArcMap venue, and all of the functions would be accessible from a customized menu. The ultimate goal is to gain the ability to carry out Topologic Navigations on watershed data that does not originally contain Pfafstetter-based attributes. Theoretically, with such data it would be possible to perform all of the steps required in the navigation with one "click of a button" on an ArcMap toolbar.

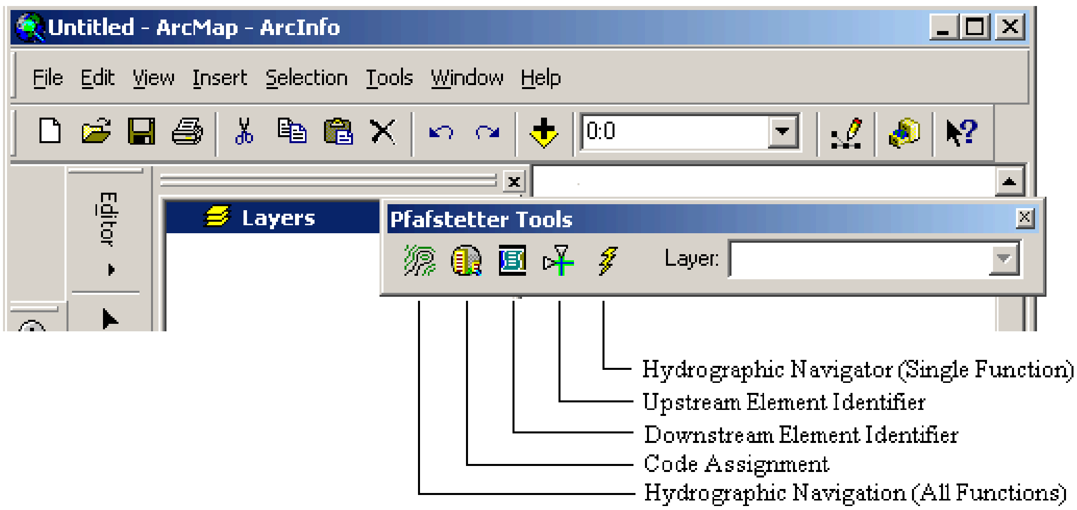


Figure 7.1 – Simulated ArcGIS toolbar for the Pfafstetter Tools

The second step in implementing the Pfafstetter Tools into the ArcGIS environment is to develop an interface from which the tools could access data stored in the format specified by the ArcGIS Hydro Data Model. With such an interface, the tools could be packaged and disseminated to all users of the ArcGIS Hydro Data Model.

7.3 General Coding Improvements

As always, computer programs may be coded in various ways, and some approaches are more efficient than others. The program code behind the Pfafstetter Tools has undergone numerous revisions, but is far from perfect. The code could be streamlined for efficiency, and it could be made to be more flexible in terms of its required data inputs. For example, the Downstream Area Identification code searches the database for fields of the form “LEVELX” where X is a user specified integer. The function will not execute if the database does not have a field of this format. The code could be adjusted so that the program generates a list of database fields and then lets the user select which field contains which data required by the macro. In this way, the data would not require such a rigid format.

The program codes for the current versions of the Downstream Area Identification, Upstream Area Identification, and Topologic Navigation macros are provided in Appendix A so that interested parties may examine and modify the code in order to make improvements.

7.4 Raster-based Downstream Code Assignments

Topologic Navigation is only dependent upon the downstream area codes, and one method for determining downstream areas is through use of Pfafstetter-based numbering systems. However, alternative methods for obtaining downstream codes are also possible. One possible method involves the use of various grids created in the watershed delineation process (Section 2.1).

The outlet grid consists of the outlet points for each drainage area in the watershed grid. These outlets are attributed with the numerical code of the cells making up their watershed. This code becomes the watershed’s ID when the watershed grid is converted into a vector format. Once a vector drainage area dataset is obtained, the outlet cells in the outlet grid could be re-numbered with the value of the cells in the downstream watershed/stream link. This could be accomplished by inspecting the value of the flow direction grid at each outlet in the outlet grid. This value determines the location of the first cell in the watershed downstream of the outlet. The value stored in this cell is therefore the code of the watershed downstream of the outlet. By attributing each outlet cell with the code of the downstream watershed/streamlink, then the resulting watershed delineation from this grid will attribute each watershed with the ID of the watershed downstream of itself in the original watershed grid. This new watershed grid could be vectorized and intersected with the original watershed dataset, and the result would be a watershed dataset with each area attributed with a unique ID and the ID of the area downstream.

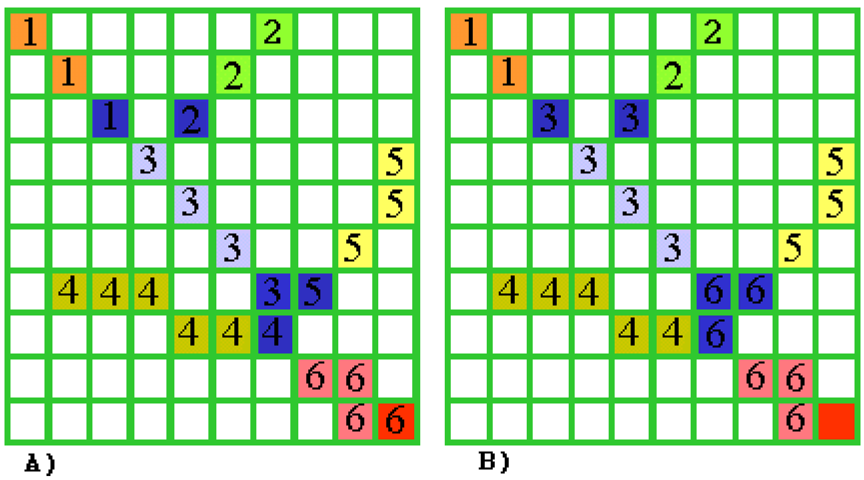


Figure 7.2 – Stream Link/Outlet Grids – A) Stream link/outlet grid, with outlets in blue. B) modified link/outlet grid with outlets taking the value of the downstream links. The overall outlet does not have a downstream link, so its outlet value is undefined (Red).

This method, if proven viable, would eliminate the necessity for use of the Pfafstetter coding system in order to describe landscape topologies. However, a secondary benefit of using the Pfafstetter coding system is that users who are familiar with the system can easily predict downstream and upstream areas without the need for a computer. For example, if a user knew a dataset was attributed with the Pfafstetter-based codes, then the user would also know that area #8934 is immediately upstream of area #8933. If this same dataset was attributed with downstream codes based on some other method, the user may not be able to manually determine the area's relative topology based solely on the area's ID code.

Appendix A

Visual Basic™ Code for Pfafstetter Tools Macros

DEFINITIONS Module

```
Option Base 1
Public basinlevel
Public Totalcode()
Public Dwnstrm()
Public Comments() As String
Public numcolumns As Integer
Public numrows As Integer
Public success As Integer
Public check As Integer
Public unknown As Integer
Public BasinName() As String
Public DwnstrmName() As String
Public UpstrmName() As String
Public Upstrm()
Public Trace()
Public basin
Public locator As Integer
Public Fnode() As Integer
Public tnode() As Integer
Public nodelist()
Public cfnode()
Public ctnode()
Public countnode
Public coastalline() As String
Public coastalpoint() As String
Public prompt1
Public text1
Public oops
Public test1
Public column As Integer
Public maxrows() As Integer
Public topfour(4, 2)

Public levels()
Public break() As Long
Public test() As Integer
```

DOWNSTREAM AREA IDENTIFICATION MACRO

```
Sub downstream()

Dim prompt
Dim text As String
Dim results As String
Dim percentSuccess
Dim percentChecked
Dim integerSuccess As Integer
Dim integerchecked As Integer

Application.Cursor = xlWait

'Initialize variables
numcolumns = 0
numrows = 0
test1 = "hello"
success = 0
unknown = 0
check = 0

'Determine the Number of Columns in the Worksheet
'This reads the value in Row 1 for each column until no value is given
'Numcolumns is incremented until the last filled column in the dbf file is reached

Do While test1 <> ""
   test1 = Cells(1, numcolumns + 1).Value
   If test1 = "" Then Exit Do
   numcolumns = numcolumns + 1
Loop

'Determine the Number of Rows in the Worksheet
'The number of rows in the dbf file will be the maximum number of rows in all columns
'The statement reads the value in each cell in the column until the value is null
'To account for possible empty or missing values in a column, the statement runs until
'the current cell and the cell 5 rows down are both null. This will skip over a few blank
'cells, but if the final 5 rows in a column are blank, they will not be counted
```

```
ReDim maxrows(numcolumns)
For I = 1 To numcolumns
   test1 = "hello"
   Do While test1 <> ""
      test1 = Cells(numrows + 1, I).Value
      If test1 = "" Then
         test1 = Cells(numrows + 6, I).Value
         If test1 = "" Then Exit Do
      End If
      numrows = numrows + 1
      maxrows(I) = numrows
   Loop
   numrows = 0
Next I
J = 0
For I = 1 To numcolumns 'The number of rows is the maximum number of rows in any given
column
   J = maxrows(I)
   If J > numrows Then
      numrows = J
   End If
Next I

Application.Cursor = xlDefault
'Get Pfafstetter Level from User - callin the level function
level
Application.Cursor = xlWait
'Find Column for processing

test1 = "hello"
column = 1
Do While test1 <> "LEVEL" & basinlevel   'It searches the column headings for LEVEL2, LEVEL3
or LEVEL4
   test1 = Cells(1, column).Value       'Depending on which level the user selected
   If test1 = "LEVEL" & basinlevel Then Exit Do
   column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
   If column = numcolumns + 1 Then 'Desired Level not in file - user must select another level
      column = 1
      Application.Cursor = xlDefault
      prompt1 = "Selected Level not found, Enter a different level"
      text1 = "Error"
      oops = MsgBox(prompt1, vbOKCancel, text1)
      If oops = vbCancel Then
         End
      End If
      level
   End If
Loop
```

```
'Write Selected level codes to totalcode array
'Numrows-1 is used so that the column header is not included in the arrays
'The first value of the Totalcode array is from the 2nd row - the column heading in row 1 is not included

ReDim Totalcode(numrows - 1)
For I = 1 To numrows - 1
   Totalcode(I) = Cells(I + 1, column).Value
Next I

'Break Pfafstetter Code into Separate Digits
digits

'Determine downstream element
ReDim Dwnstrm(numrows - 1)
ReDim Comments(numrows - 1)
For I = 1 To numrows - 1
   Dwnstrm(I) = -11111
Next I
ReDim test(basinlevel)
zeros
For I = 1 To numrows - 1
   If basinlevel = 1 And Totalcode(I) <> 0 Then
      Dwnstrm(I) = -999
      Comments(I) = "Water"
      GoTo increment
   End If
   If basinlevel = 1 And Totalcode(I) = 0 Then
      Dwnstrm(I) = -888
      Comments(I) = "Internal"
      GoTo increment
   End If
   If Totalcode(I) < 0 Then
      Dwnstrm(I) = -444
      Comments(I) = "UNKNOWN"
   End If
   classify (I)
   If Dwnstrm(I) = -11111 Then
      water (I)
      If Dwnstrm(I) <> -11111 Then
         GoTo increment
      End If
      identify (I)
      If Dwnstrm(I) <> -11111 Then
         GoTo increment
      End If
   End If
   If Dwnstrm(I) = -555 Then
      density (I)
```

```
        If Dwnstrm(I) <> -555 Then
          GoTo increment
        End If
      End If
increment:
Next I

border
three_rivers

'Write dwnstrm and Comments arrays to the spreadsheet
'dwnstrm array contains either the code of the dwnstrm basin or -888 (Internal)
'-444 (Unknown), -999 (Water), -66666 (Check), or -777 (Border)
'The comments arrray contains either "Success", "Check - Internal", "Check", "Check - Border",
'"unknown", or "water"

 Cells(1, numcolumns + 1).Select

        ActiveCell.Offset(0, 0) = "Downstream"
        ActiveCell.Offset(0, 1) = "Comments"

        For I = 1 To numrows - 1
          ActiveCell.Offset(I, 0) = Dwnstrm(I)
          ActiveCell.Offset(I, 1) = Comments(I)
          If Comments(I) <> "???" And Comments(I) <> "Check - Border" And Comments(I) <>
"Check - Internal" Then
            success = success + 1
          End If
          If Comments(I) = "???" Then
            unknown = unknown + 1
          End If
          If Comments(I) = "Check - Border" Or Comments(I) = "Check - Internal" Then
            check = check + 1
          End If
        Next I

'Adjust database range to include new fields
'This allows the file to be saved as a .dbf file
Range(Cells(1, 1), Cells(numrows, numcolumns + 2)).Name = "Database"

'Generate Summary Report
integerSuccess = ((success / (numrows - 1)) * 10000) - 0.5
integerchecked = ((check / (numrows - 1)) * 10000) - 0.5
percentSuccess = integerSuccess / 100
percentChecked = integerchecked / 100
```

```
results = "Results:" & Chr(13) & Chr(10) & "Total Determinations = " & numrows - 1 _
        & Chr(13) & Chr(10) & "Successful Determinations = " _
        & success & " (" & percentSuccess & Chr(37) & ")" & Chr(13) & Chr(10) _
        & "Basins to be checked = " & check _
        & " (" & percentChecked & Chr(37) & ")" & Chr(13) & Chr(10) & "Unknowns = " &
unknown
        text1 = "Downstream Determination Results"
        Application.Cursor = xlDefault
        oops = MsgBox(results, vbOKOnly, text1)

End Sub
```

```
Public Function level()
'Get Appropriate Level from user
'While it is theoretically possible to have Pfafstetter basins up to level 9,
'The HYDRO1K data for which this program was written is often only reliable
'Up to Level 4. (Although Level 5 and Level 6 numbers are given in the Attribute Tables
'This code was written only for proceesing Level 2, Level 3, or Level 4 data.
'If time allows, code will be written for up to 9 levels

basinlevel = -2
Do While basinlevel < 0
      text = "Pfafstetter Level Selection"
      prompt = "Enter the Pfafstetter Level:"
      basinlevel = InputBox$(prompt, text)
         If basinlevel = "" Then
            End
         End If
         basinlevel = CInt(basinlevel)
         If basinlevel < 0 Then
               prompt1 = "The Pfafstetter Level must be an integer greater than zero"
               text1 = "Invalid Selection"
               oops = MsgBox(prompt1, vbOKCancel, text1)
               If oops = vbCancel Then
                  End
               End If
         End If
Loop
End Function
```

```
Public Function digits()
Dim hold As Long
Dim hold_double As Double
Dim Jstart As Integer
Dim hold1
Dim code
Dim rounddown_index As Integer

ReDim levels(numrows - 1, basinlevel)
rounddown_index = (basinlevel / 7) + 0.5
ReDim break(rounddown_index)

For I = 1 To numrows - 1
  hold_double = Totalcode(I)
  For J = rounddown_index To 1 Step -1
    If J = 1 Then
      hold = (hold_double / (10 ^ (7 * (J - 1))))
    End If
    If J <> 1 Then
      hold = (hold_double / (10 ^ (7 * (J - 1)))) - 0.5
    End If
    break(J) = hold
    hold_double = hold_double - hold * 10 ^ (7 * (J - 1))

  Next J
  J = 1
  For K = 1 To rounddown_index
    Jstart = J
    code = break(K)
    Do While J <> basinlevel + 1 And J <> Jstart + 7
      If code < 0 Then
        levels(I, basinlevel - J + 1) = code
        code = 0
        GoTo zero
      End If
      hold1 = code
      hold = rounddown(code)
      levels(I, basinlevel - J + 1) = hold1 - hold
      code = hold / 10
zero:
      J = J + 1
    Loop
  Next K
Next I

End Function
```

```
Public Function rounddown(code)

Dim test As Single
Dim test1 As Long
Dim remainder As Single

remainder = 1
test = 1
test1 = 1

If code > 0 Then
   code = code + 1
   Do While remainder <> 0
      code = code - 1
      test = code / 10
      test1 = (code - 0.5) / 10
      remainder = test - test1
   Loop
End If

rounddown = code

End Function
```

```
Public Function classify(I)

For J = 1 To basinlevel
   If levels(I, J) = 2 Or levels(I, J) = 4 Or levels(I, J) = 6 Or levels(I, J) = 8 Then
      test(J) = 2
   End If
   If levels(I, J) = 3 Or levels(I, J) = 5 Or levels(I, J) = 7 Or levels(I, J) = 9 Then
      test(J) = 3
   End If
   If levels(I, J) = 0 Then
      test(J) = 0
   End If
   If levels(I, J) = 1 Then
      test(J) = 1
   End If
Next J

End Function
```

```
Public Function zeros()

For I = 1 To numrows - 1
   For J = 1 To basinlevel
      If levels(I, J) = 0 Then
         Dwnstrm(I) = -555
         Comments(I) = "Check - Internal"
      End If
   Next J
Next I

End Function
```

```
Public Function identify(I)

Dim max
Dim maximum
Dim count
Dim upper
Dim lower
Dim lower1
Dim index As Integer

If test(basinlevel) = 3 Then
   Dwnstrm(I) = Totalcode(I) - 2
   Comments(I) = "Success"
End If
If test(basinlevel) = 2 Then
   Dwnstrm(I) = Totalcode(I) - 1
   Comments(I) = "Success"
End If
If test(basinlevel) = 1 Then
      J = basinlevel - 1
      count = 1
      If test(J) = 1 Then
         Do While test(J) = 1 And J <> 0
            count = count + 1
            J = J - 1
         Loop
      End If
      If test(J) = 3 Then
         maximum = 0
         max = 0
         upper = 0
         lower = 0
         lower1 = 0
         For K = count To 0 Step -1
            upper = 10 ^ (K) + upper
         Next K
```

```
    If count > 1 Then
      For K = count - 1 To 1 Step -1
        If K <> -1 Then
          lower1 = (10 ^ K) + lower1
        End If
      Next K
    End If
    lower = (2 * upper) - lower1
    For K = 1 To numrows - 1
      If Totalcode(K) < Totalcode(I) - upper And Totalcode(K) > Totalcode(I) - lower Then
        max = Totalcode(K)
        If max > maximum Then
          maximum = max
          index = K
        End If
      End If
    Next K
    If maximum = 0 Then
      Dwnstrm(I) = -777
      Comments(I) = "Border"
    End If
    If maximum <> 0 Then
      classify (index)
      If test(basinlevel) <> 2 Then
        Dwnstrm(I) = maximum
        Comments(I) = "Success"
      End If
      If test(basinlevel) = 2 Then
        Dwnstrm(I) = -666
        Comments(I) = "Check - Border"
      End If
      classify (I)
    End If
  End If
  If test(J) = 2 Then
    maximum = 0
    max = 0
    upper = 0
    lower = 0
    lower1 = 0
    For K = count - 1 To -1 Step -1
      If K <> -1 Then
        upper = 10 ^ (K) + upper
      End If
    Next K
    lower = (10 * upper) + 2
    For K = 1 To numrows - 1
      If Totalcode(K) < Totalcode(I) - upper And Totalcode(K) > Totalcode(I) - lower Then
        max = Totalcode(K)
        If max > maximum Then
```

```
                maximum = max
                index = K
              End If
            End If
          Next K
          If maximum = 0 Then
            Dwnstrm(I) = -777
            Comments(I) = "Border"
          End If
          If maximum <> 0 Then
            classify (index)
            If test(basinlevel) <> 2 Then
              Dwnstrm(I) = maximum
              Comments(I) = "Success"
            End If
            If test(basinlevel) = 2 Then
              Dwnstrm(I) = -666
              Comments(I) = "Check - Border"
            End If
            classify (I)
          End If
        End If
      End If
End Function
```

```
Public Function border()

'Check to see if the determined dwnstrm basin exists in the dataset
For I = 1 To numrows - 1
  J = 1
  Do While J <> numrows
    If Dwnstrm(I) < 0 Then   'This eliminates all basins draining to the ocean
      J = numrows            'As well as those that must be checked already
    End If
    If J <> numrows Then
      Do While Dwnstrm(I) <> Totalcode(J) And J <> numrows - 1
        J = J + 1
      Loop
      If Dwnstrm(I) = Totalcode(J) Then
        J = numrows
      End If
      If J = numrows - 1 Then            'The dwnstrm basin is not in the database
        Dwnstrm(I) = -666          'It must be checked manually
        Comments(I) = "Check - Border"
        J = numrows
      End If
    End If
  Loop
Next I
End Function
```

```
Public Function water(I)

Dim odd As Boolean
Dim countone As Integer
Dim uno As Integer
Dim evenmarker As Integer

uno = 0
evenmarker = basinlevel + 1
odd = True

'Case 1, 2 - All digits are odd, except the last digit - See reference section 2-1, 2-2

For J = 1 To basinlevel - 1
   If test(J) = 1 Or test(J) = 3 Then
      odd = True
   End If
   If test(J) <> 1 And test(J) <> 3 Then
      odd = False
      GoTo oddnot
   End If
Next J
oddnot:
If odd = True Then
   Dwnstrm(I) = -999
   Comments(I) = "Water"
End If

'Case 3 - First digit is even, the rest of the digits are 1 - See reference section 2-3

countone = 0
If test(1) = 2 Then
   For J = 2 To basinlevel
      If test(J) = 1 Then
         countone = countone + 1
      End If
   Next J
End If
If countone = basinlevel - 1 And countone <> 0 Then
   Dwnstrm(I) = -999
   Comments(I) = "Water"
End If

' Case 4 - First Few digits are odd, the one digit is even, then the rest are 1 - See Reference Section
2-4

If test(basinlevel) = 1 Then
   If test(1) = 3 Or test(1) = 1 Then
      For J = 2 To basinlevel - 1
```

```
        If evenmarker > (J) Then
          If test(J) = 2 Then
            evenmarker = J
          End If
        End If
        If evenmarker < J Then
          If test(J) <> 1 Then
            uno = uno + 1
          End If
        End If
      Next J
      If evenmarker <> basinlevel + 1 And uno = 0 Then
        Dwnstrm(I) = -999
        Comments(I) = "Water"
      End If
    End If
End If
End Function
```

```
Public Function density(I)

Dim countnumber As Integer
Dim countzero As Integer
Dim nodrain As Boolean
Dim odd As Boolean
Dim countone As Integer
Dim uno As Integer
Dim evenmarker As Integer
Dim max
Dim maximum
Dim count
Dim upper
Dim lower
Dim lower1
Dim countuno As Integer
Dim zerostring As String
Dim errortext As String
Dim text1 As String


' This function considers all the areas with one or more than one "0" digit in the
' Pfafstetter code. It first determines if the area is internal, or if it is density
' lacking. See documentation

'Added 4/19/01
count = 0
J = basinlevel
Do While test(J) = 0 And J > 0
  count = count + 1
  J = J - 1
```

```
Loop

If count = basinlevel Then
   For J = 1 To numrows - 1
      If Totalcode(J) < 10 ^ count And Totalcode(J) > Totalcode(I) Then
         Dwnstrm(I) = -888
         Comments(I) = "Internal"
         GoTo incrementdensity
      End If
   Next J
   zerostring = "0"
   For p = 2 To basinlevel
      zerostring = zerostring & "0"
   Next p
   errortext = "All areas have the code " & zerostring & ", code = " & Totalcode(I) & count &
basinlevel
   text1 = "Downstream Determination Results"
   Application.Cursor = xlDefault
   oops = MsgBox(errortext, vbOKOnly, text1)
   End
End If

If basinlevel - count - 1 > 0 Then
   If test(basinlevel - count - 1) = 0 Then
      Dwnstrm(I) = -888
      Comments(I) = "Internal"
      GoTo incrementdensity
   End If
End If

If test(basinlevel - count) = 1 Then
   countone = 1
   J = basinlevel - count - 1
   Do While J > 0 And test(J) = 1
     J = J - 1
     countone = countone + 1
   Loop
   If basinlevel - count - countone = 0 Then
      For J = 1 To numrows - 1
         If Totalcode(J) < Totalcode(I) + 10 ^ count And Totalcode(J) > Totalcode(I) Then
            Dwnstrm(I) = -888
            Comments(I) = "Internal"
            GoTo incrementdensity
         End If
      Next J
      Dwnstrm(I) = -999
      Comments(I) = "Water"
      GoTo incrementdensity
   End If
   If test(basinlevel - count - countone) = 0 Then
```

```
      Dwnstrm(I) = -888
      Comments(I) = "Internal"
      GoTo incrementdensity
   End If
   If basinlevel - count - countone - 1 > 0 Then
      If test(basinlevel - count - countone - 1) = 0 Then
         Dwnstrm(I) = -888
         Comments(I) = "Internal"
         GoTo incrementdensity
      End If
   End If
End If


'Check for density problems
If count > 0 Then
   For J = 1 To numrows - 1
      If Totalcode(J) < Totalcode(I) + 10 ^ count And Totalcode(J) > Totalcode(I) Then
         Dwnstrm(I) = -888
         Comments(I) = "Internal"
         GoTo incrementdensity
      End If
   Next J
End If

'To make it to this point, an area must have a density problem, or it's drainage is discernible
'by the identify functions. If count = 0, then the drainage is discernible with the identify functions
'and the area will not drain to the ocean. If count > 0, then area either drains to ocean or to another
basin

'Test for draining to the ocean using nonzero digits of totalcode(I)
uno = 0
evenmarker = basinlevel + 1
odd = True

'Case 1, 2 - All digits are odd, except the last digit

For J = 1 To basinlevel - count - 1
   If test(J) = 1 Or test(J) = 3 Then
      odd = True
   End If
   If test(J) <> 1 And test(J) <> 3 Then
      odd = False
      GoTo notodd
   End If
Next J
notodd:
If odd = True Then
   Dwnstrm(I) = -999
   Comments(I) = "Water"
```

```
   GoTo incrementdensity
End If


'Case 3 - First digit is even, the rest of the digits are 1

countone = 0
If test(1) = 2 Then
   For J = 2 To basinlevel - count
      If test(J) = 1 Then
         countone = countone + 1
      End If
   Next J
End If
If countone = basinlevel - count - 1 Then
   Dwnstrm(I) = -999
   Comments(I) = "Water"
   GoTo incrementdensity
End If

' Case 4 - First Few digits are odd, then one digit is even, then the rest are 1 –

If test(basinlevel - count) = 1 Then
   If test(1) = 3 Or test(1) = 1 Then
      For J = 2 To basinlevel - count - 1
         If evenmarker > (J) Then
            If test(J) = 2 Then
               evenmarker = J
            End If
         End If
         If evenmarker < J Then
            If test(J) <> 1 Then
               uno = uno + 1
            End If
         End If
      Next J
      If evenmarker <> basinlevel + 1 And uno = 0 Then
         Dwnstrm(I) = -999
         Comments(I) = "Water"
         GoTo incrementdensity
      End If
   End If
End If
If test(basinlevel - count) = 3 Then
   maximum = 0
   max = 0
   For J = 1 To numrows - 1
      If Totalcode(J) < Totalcode(I) - 10 ^ count And Totalcode(J) > Totalcode(I) - (2 * (10 ^ count))
- 1 Then
         max = Totalcode(J)
```

```
      If max > maximum Then
        maximum = max
      End If
    End If
  Next J
  If maximum = 0 Then
    Dwnstrm(I) = -777
    Comments(I) = "Border"
    GoTo incrementdensity
  End If
  If maximum <> 0 Then
    Dwnstrm(I) = maximum
    Comments(I) = "Success"
    GoTo incrementdensity
  End If
End If
If test(basinlevel - count) = 2 Then
  maximum = 0
  max = 0
  For J = 1 To numrows - 1
    If Totalcode(J) < Totalcode(I) And Totalcode(J) > Totalcode(I) - (10 ^ count) - 1 Then
      max = Totalcode(J)
      If max > maximum Then
        maximum = max
      End If
    End If
  Next J
  If maximum = 0 Then
    Dwnstrm(I) = -777
    Comments(I) = "Border"
    GoTo incrementdensity
  End If
  If maximum <> 0 Then
    Dwnstrm(I) = maximum
    Comments(I) = "Success"
    GoTo incrementdensity
  End If
End If
If test(basinlevel - count) = 1 Then
  If test(basinlevel - count - 1) = 2 Then
    maximum = 0
    max = 0
    lower = (10 ^ count) + (10 ^ (count + 1)) + 1
    For J = 1 To numrows - 1
      If Totalcode(J) < (Totalcode(I) - (10 ^ count)) And Totalcode(J) > Totalcode(I) - lower Then
        max = Totalcode(J)
        If max > maximum Then
          maximum = max
        End If
      End If
```

```
    Next J
    If maximum = 0 Then
      Dwnstrm(I) = -777
      Comments(I) = "Border"
      GoTo incrementdensity
    End If
    If maximum <> 0 Then
      Dwnstrm(I) = maximum
      Comments(I) = "Success"
      GoTo incrementdensity
    End If
  End If
  If test(basinlevel - count - 1) = 3 Then
    maximum = 0
    max = 0
    upper = (10 ^ count) + (10 ^ (count + 1))
    lower = (10 ^ count) + 2 * (10 ^ (count + 1)) + 1
    For J = 1 To numrows - 1
      If Totalcode(J) < Totalcode(I) - upper And Totalcode(J) > Totalcode(I) - lower Then
        max = Totalcode(J)
        If max > maximum Then
          maximum = max
        End If
      End If
    Next J
    If maximum = 0 Then
      Dwnstrm(I) = -777
      Comments(I) = "Border"
      GoTo incrementdensity
    End If
    If maximum <> 0 Then
      Dwnstrm(I) = maximum
      Comments(I) = "Success"
      GoTo incrementdensity
    End If
  End If
  If test(basinlevel - count - 1) = 1 Then
    J = basinlevel - count - 2
    countuno = 0
    Do While test(J) = 1
      countuno = countuno + 1
      J = J - 1
    Loop
    If test(J) = 3 Then
      maximum = 0
      max = 0
      upper = 0
      lower = 0
      lower1 = 0
      For K = countuno + count + 2 To count Step -1
```

```
      upper = 10 ^ (K) + upper
    Next K
    For K = count To count + countuno + 1
      lower1 = (10 ^ K) + lower1
    Next K
    lower = (2 * upper) - lower1 - 1
    For K = 1 To numrows - 1
      If Totalcode(K) < Totalcode(I) - upper And Totalcode(J) > Totalcode(I) - lower Then
        max = Totalcode(K)
        If max > maximum Then
          maximum = max
        End If
      End If
    Next K
    If maximum = 0 Then
      Dwnstrm(I) = -777
      Comments(I) = "Border"
      GoTo incrementdensity
    End If
    If maximum <> 0 Then
      Dwnstrm(I) = maximum
      Comments(I) = "Success"
      GoTo incrementdensity
    End If
  End If
  If test(J) = 2 Then
    maximum = 0
    max = 0
    upper = 0
    lower = 0
    lower1 = 0
    For K = count + countuno + 2 To count Step -1
      upper = 10 ^ (K) + upper
    Next K
    lower = (10 * upper) + (10 ^ count) + 1
    For K = 1 To numrows - 1
      If Totalcode(K) < Totalcode(I) - upper And Totalcode(J) > Totalcode(I) - lower Then
        max = Totalcode(K)
        If max > maximum Then
          maximum = max
        End If
      End If
    Next K
    If maximum = 0 Then
      Dwnstrm(I) = -777
      Comments(I) = "Border"
      GoTo incrementdensity
    End If
    If maximum <> 0 Then
      Dwnstrm(I) = maximum
```

```
            Comments(I) = "Success"
            GoTo incrementdensity
         End If
      End If
   End If
End If
incrementdensity:
End Function
```

```
Public Function three_rivers()

'this function will handle the rare situation in which within a given level, three rivers meet in
'one confluence.  It also definitively determines basins
'on the study area border

For I = 1 To numrows - 1
   If Dwnstrm(I) = -777 Or Dwnstrm(I) = -666 Then
      If levels(I, basinlevel) < 4 Then
         Dwnstrm(I) = -777
         Comments(I) = "Border"
      End If
      If levels(I, basinlevel) = 5 Or _
         levels(I, basinlevel) = 7 Or _
         levels(I, basinlevel) = 9 Then
            J = 1
            Do While J < numrows - 1 And Totalcode(J) <> Totalcode(I) - 1
               J = J + 1
            Loop
            If Totalcode(I) - 1 = Totalcode(J) Then
               K = 1
               Do While K < numrows - 1 And Totalcode(K) <> Totalcode(I) - 3
                  K = K + 1
               Loop
               If Totalcode(I) - 3 = Totalcode(K) Then
                  If Dwnstrm(K) = Totalcode(I) - 4 Then
                     L = 1
                     Do While L < numrows - 1 And Totalcode(L) <> Totalcode(I) - 4
                        L = L + 1
                     Loop
                     If Totalcode(L) = Totalcode(I) - 4 Then
                        If Dwnstrm(L) <> -777 And Dwnstrm(L) <> -666 Then
                           Dwnstrm(I) = Totalcode(I) - 4
                           Comments(I) = "Success"
                        End If
                        If Dwnstrm(L) = -777 Or Dwnstrm(L) = -666 Then
                           Dwnstrm(I) = -777
                           Comments(I) = "Border"
                        End If
                     End If
                     If L = numrows - 1 And Totalcode(L) <> Totalcode(I) - 4 Then
                        Dwnstrm(I) = -777
```

```
                    Comments(I) = "Border"
                End If
            End If
            If Dwnstrm(K) = -777 Or Dwnstrm(K) = -666 Then
                Dwnstrm(I) = -777
                Comments(I) = "Border"
            End If
        End If
        If K = numrows - 1 And Totalcode(K) <> Totalcode(I) - 3 Then
            Dwnstrm(I) = -777
            Comments(I) = "Border"
        End If
    End If
    If J = numrows - 1 And Totalcode(J) <> Totalcode(I) - 1 Then
        Dwnstrm(I) = -777
        Comments(I) = "Border"
    End If
End If
If levels(I, basinlevel) = 4 Or _
   levels(I, basinlevel) = 6 Or _
   levels(I, basinlevel) = 8 Then
    J = 1
    Do While J < numrows - 1 And Totalcode(J) <> Totalcode(I) + 1
        J = J + 1
    Loop
    If Totalcode(I) + 1 = Totalcode(J) Then
        If Dwnstrm(J) = Totalcode(I) - 3 Or Dwnstrm(J) = -777 Or Dwnstrm(I) = -666 Then
            K = 1
            Do While K < numrows - 1 And Totalcode(K) <> Totalcode(I) - 2
                K = K + 1
            Loop
            If Totalcode(I) - 2 = Totalcode(K) Then
                If Dwnstrm(K) = Totalcode(I) - 3 Then
                    L = 1
                    Do While L < numrows - 1 And Totalcode(L) <> Totalcode(I) - 3
                        L = L + 1
                    Loop
                    If Totalcode(L) = Totalcode(I) - 3 Then
                        If Dwnstrm(L) <> -777 And Dwnstrm(L) <> -666 Then
                            Dwnstrm(I) = Totalcode(I) - 3
                            Comments(I) = "Success"
                        End If
                        If Dwnstrm(L) = -777 Or Dwnstrm(L) = -666 Then
                            Dwnstrm(I) = -777
                            Comments(I) = "Border"
                        End If
                    End If
                    If L = numrows - 1 And Totalcode(L) <> Totalcode(I) - 3 Then
                        Dwnstrm(I) = -777
                        Comments(I) = "Border"
```

```
                    End If
                  End If
                  If Dwnstrm(K) = -777 Or Dwnstrm(K) = -666 Then
                    Dwnstrm(I) = -777
                    Comments(I) = "Border"
                  End If
                End If
                If K = numrows - 1 And Totalcode(K) <> Totalcode(I) - 2 Then
                  Dwnstrm(I) = -777
                  Comments(I) = "Border"
                End If
              End If
            End If
            If J = numrows - 1 And Totalcode(J) <> Totalcode(I) + 1 Then
              Dwnstrm(I) = -777
              Comments(I) = "Border"
            End If
      End If
    End If
Next I

End Function
```

UPSTREAM AREA IDENTIFICATION MACRO

```
Sub upstream()

Dim prompt
Dim text As String
Dim count As Integer
Dim maxcount As Integer
Dim test

numcolumns = 0
numrows = 0
test = "jordan"
Application.Cursor = xlWait
'Determine the Number of Columns in the Worksheet
'This reads the value in Row 1 for each column until no value is given
'Numcolumns is incremented until the last filled column in the dbf file is reached

Do While test <> ""
   test = Cells(1, numcolumns + 1).Value
   If test = "" Then Exit Do
   numcolumns = numcolumns + 1
Loop

'Determine the Number of Rows in the Worksheet
'The number of rows in the dbf file will be the maximum number of rows in all columns
'The statement reads the value in each cell in the column until the value is null
'To account for possible empty or missing values in a column, the statement runs until
'the current cell and the cell 5 rows down are both null. This will skip over a few blank
'cells, but if the final 5 rows in a column are blank, they will not be counted

ReDim maxrows(numcolumns)
For I = 1 To numcolumns
   test = "hello"
   Do While test <> ""
      test = Cells(numrows + 1, I).Value
      If test = "" Then
         test = Cells(numrows + 6, I).Value
         If test = "" Then Exit Do
      End If
      numrows = numrows + 1
      maxrows(I) = numrows
   Loop
   numrows = 0
Next I
```

```
J = 0
For I = 1 To numcolumns 'The number of rows is the maximum number of rows in any column
    J = maxrows(I)
    If J > numrows Then
        numrows = J
    End If
Next I
Application.Cursor = xlDefault
'Get Pfafstetter Level from User - calling the level function
level
Application.Cursor = xlWait
'Find Column containing TotalCodes

test = "hello"
column = 1
Do While test <> "LEVEL" & basinlevel   'It searches the column headings for LEVEL2, LEVEL3
or LEVEL4
    test = Cells(1, column).Value       'Depending on which level the user selected
    If test = "LEVEL" & basinlevel Then Exit Do
    column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
    If column = numcolumns + 1 Then 'Desired Level not in file - user must select another level
        column = 1
        Application.Cursor = xlDefault
        prompt1 = "Selected Level not found, Enter a different level"
        text1 = "Error"
        oops = MsgBox(prompt1, vbOKCancel, text1)
        If oops = vbCancel Then
            End
        End If
        level
    End If
Loop

'Write Selected level codes to totalcode array
'Numrows-1 is used so that the column header is not included in the arrays
'The first value of the Totalcode array is from the 2nd row - the column heading in row 1 is not
included

ReDim Totalcode(numrows - 1)
For I = 1 To numrows - 1
    Totalcode(I) = Cells(I + 1, column).Value
Next I

'Find Column containing downstream Codes

test = "hello"
column = 1
Do While test <> "DOWNSTREAM" And test <> "Downstream"  'It searches the column headings
for LEVEL2, LEVEL3 or LEVEL4
```

```
   test = Cells(1, column).Value       'Depending on which level the user selected
   If test = "DOWNSTREAM" Or test = "Downstream" Then Exit Do
   column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
   If column = numcolumns + 1 Then 'Desired Level not in file - user must select another level
      column = 1
      Application.Cursor = xlDefault
      prompt1 = "Downstream Codes are not found, Run the Downstream Element Identifier"
      text1 = "Error"
      oops = MsgBox(prompt1, vbOKOnly, text1)
      End
   End If
Loop

'Write Downstream codes to dwnstrm array
'Numrows-1 is used so that the column header is not included in the arrays
'The first value of the dwnstrm array is from the 2nd row - the column heading in row 1 is not
included

ReDim Dwnstrm(numrows - 1)
For I = 1 To numrows - 1
   Dwnstrm(I) = Cells(I + 1, column).Value
Next I

'Break Pfafstetter Code into Separate Digits
digits

'Determine upstream element
up_identify
End Sub
```

```
Public Function up_identify()

Dim count
Dim maxcount
Dim oops
Dim prompt1 As String
Dim text1 As String

maxcount = 0
For I = 1 To numrows - 1
   If levels(I, basinlevel) = 0 Or _
      levels(I, basinlevel) = 1 Or _
      levels(I, basinlevel) = 3 Or _
      levels(I, basinlevel) = 5 Or _
      levels(I, basinlevel) = 7 Or _
      levels(I, basinlevel) = 9 Then
         If Dwnstrm(I) <> -99999 And Totalcode(I) <> -1 Then
            J = 1
            count = 0
            Do While J <> numrows - 1
               If Totalcode(I) = Dwnstrm(J) Then
                  count = count + 1
               End If
               If count > maxcount Then
                  maxcount = count
               End If
               J = J + 1
            Loop
         End If
   End If
Next I

ReDim Upstrm(numrows - 1, maxcount)
For I = 1 To numrows - 1
   If levels(I, basinlevel) = 0 Or _
      levels(I, basinlevel) = 1 Or _
      levels(I, basinlevel) = 3 Or _
      levels(I, basinlevel) = 5 Or _
      levels(I, basinlevel) = 7 Or _
      levels(I, basinlevel) = 9 Then
         If Dwnstrm(I) <> -99999 And Totalcode(I) <> -1 Then
            J = 1
            count = 0
            Do While J <> numrows - 1
               If Totalcode(I) = Dwnstrm(J) Then
                  count = count + 1
                  Upstrm(I, count) = Totalcode(J)
               End If
               J = J + 1
```

```
            Loop
         End If
      End If
Next I
'print upstream info to spreadsheet, adjust database file accordingly

 Cells(1, numcolumns).Select

      If maxcount <> 0 Then
         For I = 1 To maxcount
            ActiveCell.Offset(0, I) = "Upstream" & I
            For J = 1 To numrows - 1
               ActiveCell.Offset(J, I) = Upstrm(J, I)
            Next J
         Next I
      End If

      If maxcount = 0 Then
         text1 = "Notice:"
         Application.Cursor = xlDefault
         prompt1 = "All basins drain either to the ocean or are internal"
         oops = MsgBox(prompt1, vbOKOnly, text1)
      End If


'Adjust database range to include new fields
'This allows the file to be saved as a .dbf file
Range(Cells(1, 1), Cells(numrows, numcolumns + maxcount)).Name = "Database"

'Program completion message
         text1 = "Upstream Identification Complete!"
         Application.Cursor = xlDefault
         prompt1 = "The Upstream Determination was successful"
         oops = MsgBox(prompt1, vbOKOnly, text1)

End Function
```

TOPOLOGIC NAVIGATION MACRO

```
Sub network_trace()

'With this function, the user will be able to perform upstream and downstream traces
'from a selected basin. It is desireable that the user be able to input either the
'basin number or the basin name in order to begin the trace. This will likely require
'a form with two radial buttons - I will have to figure out how to make this work.

Dim up As Integer
Dim down As Integer

Dim test

Dim ratio
up = 0
down = 0
locator = 0
basin = 0
numcolumns = 0
numrows = 0
test = "jordan"

Application.Cursor = xlWait
'Determine the Number of Columns in the Worksheet
'This reads the value in Row 1 for each column until no value is given
'Numcolumns is incremented until the last filled column in the dbf file is reached

Do While test <> ""
    test = Cells(1, numcolumns + 1).Value
    If test = "" Then Exit Do
    numcolumns = numcolumns + 1
Loop


'Determine the Number of Rows in the Worksheet
'The number of rows in the dbf file will be the maximum number of rows in all columns
'The statement reads the value in each cell in the column until the value is null
'To account for possible empty or missing values in a column, the statement runs until
'the current cell and the cell 5 rows down are both null. This will skip over a few blank
'cells, but if the final 5 rows in a column are blank, they will not be counted

ReDim maxrows(numcolumns)
For I = 1 To numcolumns
    test = "hello"
    Do While test <> ""
        test = Cells(numrows + 1, I).Value
        If test = "" Then
            test = Cells(numrows + 6, I).Value
```

```
      If test = "" Then Exit Do
    End If
    numrows = numrows + 1
    maxrows(I) = numrows
  Loop
  numrows = 0
Next I
J = 0
For I = 1 To numcolumns 'The number of rows is the maximum number of rows in any given
column
  J = maxrows(I)
  If J > numrows Then
    numrows = J
  End If
Next I
Application.Cursor = xlDefault
'get the basin level from the user
level
Application.Cursor = xlWait
'First determine if the appropriate level is in the database
test = "hello"
column = 1
Do While test <> "LEVEL" & basinlevel   'It searches the column headings for LEVEL and
basinlevel
  test = Cells(1, column).Value       'Depending on which basin the user selected
  If test = "LEVEL" & basinlevel Then Exit Do
  column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
  If column = numcolumns + 1 Then 'Desired Level not in file - user must select another basin
    column = 1
    prompt1 = "Selected basin not found, Enter a different basin"
    text1 = "Error"
    oops = MsgBox(prompt1, vbOKCancel, text1)
    If oops = vbCancel Then
      End
    End If
    level
  End If
Loop

'Get the starting basin from the user.
user:
trace_start

'Determine if input is valid
test1 = -17
I = 1
ratio = -2
Do While ratio <> 1 And I <> numrows - 1 'It searches the rows for the basin selected by the user
  I = I + 1
  test1 = Cells(I, column).Value
```

```
   ratio = test1 / basin
Loop
If ratio = 1 Then
   locator = I
End If
If I = numrows - 1 And test1 <> basin Then
   Application.Cursor = xlDefault
   prompt1 = "Selected area not found, Enter a different area's code"
   text1 = "Error"
   oops = MsgBox(prompt1, vbOKCancel, text1)
   If oops = vbCancel Then
      End
   End If
   GoTo user
End If


'Write Selected level codes to totalcode array
'Numrows-1 is used so that the column header is not included in the arrays
'The first value of the Totalcode array is from the 2nd row - the column heading in row 1 is not
included

ReDim Totalcode(numrows - 1)
For I = 1 To numrows - 1
   Totalcode(I) = Cells(I + 1, column).Value
Next I

'Find Column containing downstream Codes

test = "hello"
column = 1
Do While test <> "DOWNSTREAM" And test <> "Downstream"  'It searches the column headings
for LEVEL2, LEVEL3 or LEVEL4
   test = Cells(1, column).Value       'Depending on which level the user selected
   If test = "DOWNSTREAM" Or test = "Downstream" Then Exit Do
   column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
   If column = numcolumns + 1 Then 'Desired Level not in file - user must select another level
      column = 1
      prompt1 = "Downstream Codes are not found, Run the Downstream Element Identifier"
      text1 = "Error"
      oops = MsgBox(prompt1, vbOKOnly, text1)
      End
   End If
Loop

'Write Downstream codes to dwnstrm array
'Numrows-1 is used so that the column header is not included in the arrays
'The first value of the dwnstrm array is from the 2nd row - the column heading in row 1 not included

ReDim Dwnstrm(numrows - 1)
```

```
For I = 1 To numrows - 1
   Dwnstrm(I) = Cells(I + 1, column).Value
Next I



'Perform downstream trace
trace_down

'perform downstream trace
trace_up

Trace(locator - 1) = 1
For I = 1 To numrows - 1
   If Trace(I) <> 1 And Trace(I) <> 2 And Trace(I) <> 3 Then
      Trace(I) = 0
   End If
Next I

'Print output to spreadsheet
Cells(1, numcolumns).Select
ActiveCell.Offset(0, 1) = "Trace" & basin
For I = 1 To numrows - 1
   ActiveCell.Offset(I, 1) = Trace(I)
   If Trace(I) = 3 Then
      up = up + 1
   End If
   If Trace(I) = 2 Then
      down = down + 1
   End If

Next I

'Adjust database range to include new fields
'This allows the file to be saved as a .dbf file
Range(Cells(1, 1), Cells(numrows, numcolumns + 1)).Name = "Database"

'Program completion message
        text1 = "Area-to-Area Navigation Complete!"
        Application.Cursor = xlDefault
        prompt1 = "The area-to-area navigation from basin number " & basin & _
        " was successful" & Chr(13) & Chr(10) & Chr(13) & Chr(10) & _
        "Navigation Results:" & Chr(13) & Chr(10) & Chr(13) & Chr(10) & _
        "Downstream Basins = " & down & Chr(13) & Chr(10) & _
        "Upstream Basins = " & up
        oops = MsgBox(prompt1, vbOKOnly, text1)

End Sub
```

```
 Public Function trace_down()

Dim test As Integer
Dim prompt1 As String
Dim text1 As String
Dim oops

ReDim Trace(numrows - 1)
test = locator - 1
I = 1
If Dwnstrm(test) = -999 Then
   prompt1 = "Selected Basin drains to the ocean"
   text1 = "Notice:"
   oops = MsgBox(prompt1, vbOKOnly, text1)
End If
If Dwnstrm(test) = -888 Then
   prompt1 = "Selected Basin is an internal basin"
   text1 = "Notice:"
   oops = MsgBox(prompt1, vbOKOnly, text1)
End If
If Dwnstrm(test) = -777 Then
   prompt1 = "Selected Basin drains the study area border"
   text1 = "Notice:"
   oops = MsgBox(prompt1, vbOKOnly, text1)
End If
Do While I < numrows - 1 And Dwnstrm(test) <> -999 And Dwnstrm(test) <> -888 And
Dwnstrm(test) <> -777
   If Dwnstrm(test) = Totalcode(I) Then
      Trace(I) = 2
      test = I
      I = 0
   End If
   I = I + 1
Loop
If Dwnstrm(test) = -999 And I < numrows - 1 And test <> locator Then
   Trace(test) = 2
End If
If I = numrows - 1 And Dwnstrm(test) <> -999 And Dwnstrm(test) <> -888 And Dwnstrm(test) <> -
777 Then
   prompt1 = "Downstream Elements not identified correctly"
   text1 = "Error"
   oops = MsgBox(prompt1, vbOKCancel, text1)
   End
End If
End Function
```

```
Public Function trace_up()

Dim upcount As Integer
Dim test As String
Dim column As Integer
Dim prompt1 As String
Dim text1 As String
Dim oops
Dim test1 As Integer
Dim pointer()
Dim checked() As String
Dim check1() As Integer


'Find the columns containing the upstream codes - first check if they are there
'Then determine how many of them there are
test = "hello"
column = 1
Do While test <> "UPSTREAM1" And test <> "Upstream1"  'It searches the column headings for
LEVEL2, LEVEL3 or LEVEL4
   test = Cells(1, column).Value
   If test = "UPSTREAM1" Or test = "Upstream1" Then Exit Do
   column = column + 1 'Column is the number of the column containing the Pfafstetter Codes
   If column = numcolumns + 1 Then 'Desired Level not in file - user must select another level
      column = 1
      prompt1 = "Upstream Codes are not found, Run the Upstream Element Identifier"
      text1 = "Error"
      oops = MsgBox(prompt1, vbOKOnly, text1)
      End
   End If
Loop
test = "Upstream1"
upcount = 1
Do While test = "Upstream" & upcount Or test = "UPSTREAM" & upcount
   test = Cells(1, column + upcount).Value
   upcount = upcount + 1
Loop
upcount = upcount - 1

'Write upstream codes to upstrm array
ReDim Upstrm(numrows - 1, upcount)
For I = 1 To numrows - 1
   For J = 1 To upcount
      Upstrm(I, J) = Cells(I + 1, column + J - 1).Value
   Next J
Next I

ReDim check1(numrows - 1)
ReDim checked(numrows - 1)
```

```
ReDim pointer(upcount)

'initialize variable arrays
For I = 1 To numrows - 1
    checked(I) = "NO"
    check1(I) = 0
Next I

test1 = locator - 1
If Upstrm(test1, 1) = "" Then 'There aren't any basins upstream
    prompt1 = "There aren't any basins upstream of basin " & basin
    text1 = "Notice:"
    oops = MsgBox(prompt1, vbOKOnly, text1)
    GoTo finish
End If

If Upstrm(test1, 1) <> "" Then 'There are upstream basins
    check1(test1) = 1
    For I = 1 To numrows - 1
        If check1(I) = 1 And checked(I) <> "yes" Then
            For J = 1 To upcount
                pointer(J) = Upstrm(I, J)
                For K = 1 To numrows - 1
                    If pointer(J) <> "" And Totalcode(K) = pointer(J) Then
                        check1(K) = 1
                        Trace(K) = 3
                        K = numrows - 1
                    End If
                Next K
            Next J
            checked(I) = "yes"
            I = 0
        End If
    Next I
End If
finish:
End Function
```

```
Public Function trace_start()
Application.Cursor = xlDefault
'Get Starting Basin from user
Dim text As String
Dim prompt As String
basin = ""
text = "Hydrographic Navigation - Initital Target Area Selection"
prompt = "Enter the code of the navigation origin:"
basin = InputBox$(prompt, text)
Application.Cursor = xlWait
End Function
```

Appendix B
Upstream Area Determination Based on Pfafstetter Codes

Using the Pfafstetter System, it is fairly easy to identify the drainage area immediately upstream of a given area. In this discussion, the term "upstream" refers to the elements immediately upstream of a given element, and not all of the elements upstream of that element. The entire set of elements upstream of a given element is determined with the Topologic Navigation macro, described in Section 5.4. The individual upstream areas are determined with the Upstream Area Identification macro. The program codes for these macros are given in the Appendix A.

In most cases, there will exist two or zero drainage areas upstream of a given Pfafstetter drainage area. However, it is possible to have more than two upstream areas. These characteristics of Pfafstetter drainage areas arise because such areas are delineated from confluences or outlet points. At any confluence, at least two rivers join together, essentially combining the flows from their respective drainage areas. Therefore, the drainage area downstream of the confluence receives flow from at least two upstream drainage areas. It is possible that 3 or more rivers converge at a single confluence, thereby combining flows from 3 or more areas. This possibility is taken into consideration by the Upstream Area Identification macro.

In general, the following statements are true for Pfafstetter drainage areas, as implemented in the HYDRO1K dataset:

- Basins do not have upstream drainage areas
- Internal basins do not have upstream drainage areas
- Interbasins may have either 0, 2, or greater than 2 upstream drainage areas
- It is impossible to have 1 and only one upstream drainage area

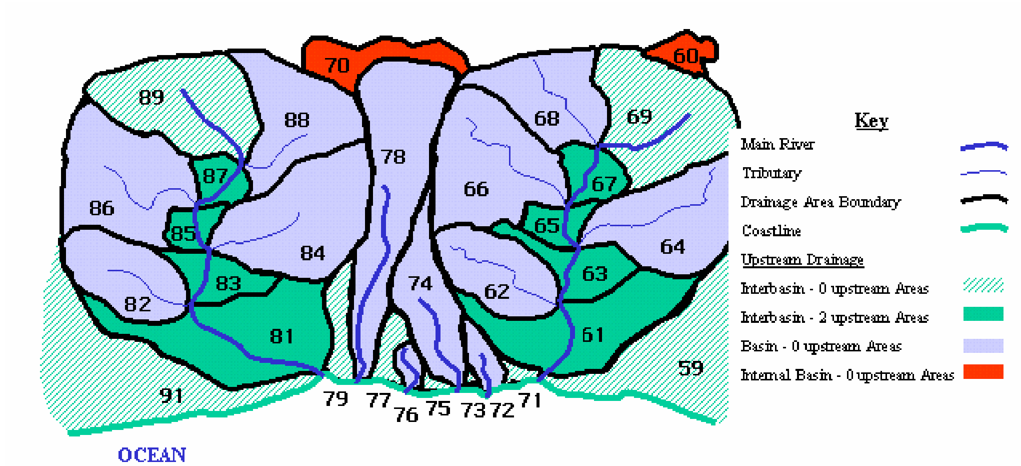


Figure B-1 – Upstream Drainage Characteristics of Pfafstetter Drainage Areas

By definition, basins do not receive flow from other drainage areas, and therefore will not have any upstream drainage areas. Basins all have even numbered Pfafstetter codes, and it is accurate to say that an even Pfafstetter code pertains to a drainage area without upstream drainage areas. Internal basins also have even Pfafstetter codes (since they end in "0"), and they also do not have any upstream drainage areas as implemented in the HYDRO1K dataset.

Interbasins drain directly to main river segments or to coastline segments, and therefore may have multiple upstream areas. Most interbasins drain to river segments, and are bounded at the upstream end by a confluence. This confluence is the outlet point for at least one basin and one distinct interbasin. Therefore most interbasins will have at least two upstream drainage areas. Interbasins draining to coastline segments are an exception to this generalization, because they will not have any upstream areas (Figure B-1). Interbasins draining to main river headwater segments will also not have upstream drainage areas. For such interbasins, their assigned Pfafstetter code will end in "9." However, this alone is not a defining characteristic for a “zero-upstream area” interbasin because a Pfafstetter code ending in “9” may be a subdivision of a lower level interbasin. In this case, the “9” interbasin would receive flow from two upstream areas. This is shown in Figure B-2 with interbasins 669 and 689. Fortunately, as noted in the description of the Downstream Area Identification methodology (Section 4.3), it is possible to distinguish coastline-draining interbasins from other interbasins. It is also easy to identify whether interbasins ending in “9” drain to main river headwater segments or to non-headwater main river segments. This is possible by inspecting the 2nd highest-level digit in the Pfafstetter code. If this digit is even, the interbasin drains to a main river headwater segment and lacks any upstream areas. If it is odd, the interbasin drains to a main river segment and receives flow from at least two upstream areas.

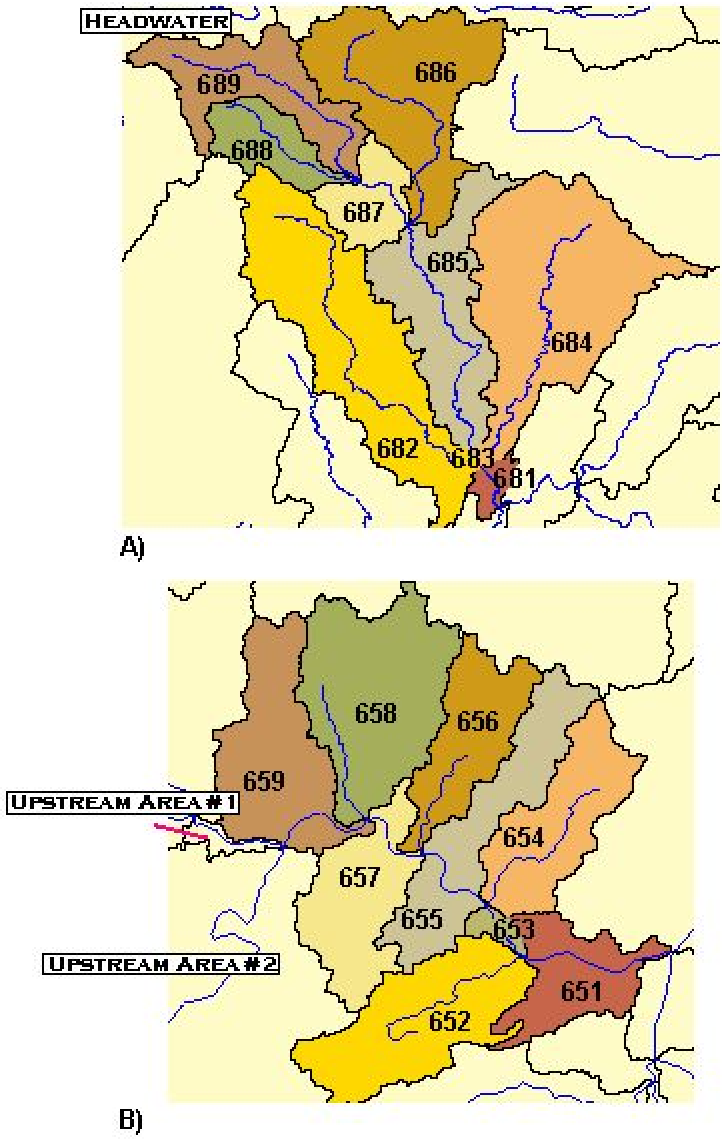


Figure B-2 –Upstream areas from "9" Interbasins –A) 0 – upstream areas from interbasin 689, B) 2- upstream areas from Interbasin 659

The steps in determining the upstream areas, including determining the number of upstream areas, would be as follows (for Pfafstetter coded data):

- Determine if the area drains to a coastline segment; If so, then there are no upstream areas

- Determine if the area is a basin (or internal basin) by checking if the Pfafstetter code is even. If so, then there are no upstream areas.
- If the area is an interbasin with a code ending in “1,” “3,” “5,” or “7,” then the two upstream areas are those with the Pfafstetter codes equal to the code of the interbasin in question, plus 1 and plus 2.
- If the area is an interbasin with a code ending in “9,” the penultimate digit becomes important. If it is even, then there are not any upstream areas. If this digit is odd, then the upstream areas are those with the Pfafstetter codes equal to the code of the interbasin in question, plus 12 and plus 2. However, if the penultimate digit is also “9,” the third highest-level digit must be examined in a similar fashion. In such a case, the upstream areas would have codes equal to the code of the interbasin in question, plus 12 or 22.
- If the area is an interbasin with three upstream areas, the upstream areas will have codes equal to the equal to the code of the interbasin in question, plus 1, 2 and 4. However, if 4 upstream areas exited, there codes would be equal to the equal to the code of the interbasin in question, plus 1, 3, 5 and 6.

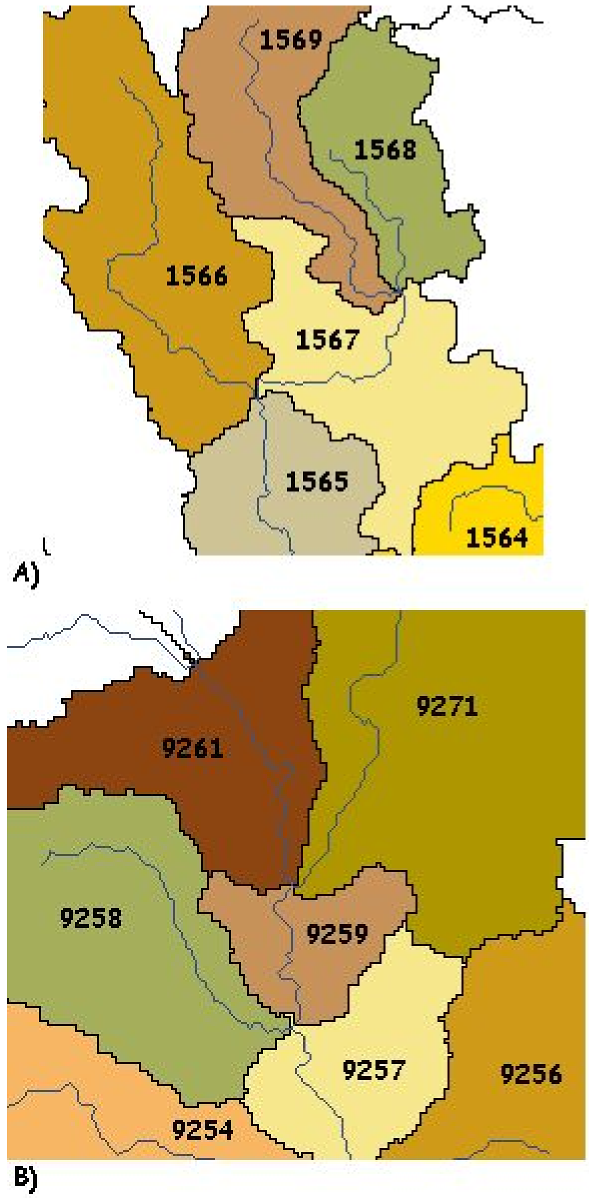


Figure B-3 –Upstream area determination –A) Because the 3rd level digit is even, interbasin 1569 does not have upstream areas. Focusing on interbasins 1565 and 1567, the upstream areas have codes equal to the area code+1 and the area code+2, B) Since the 3rd level digit is odd, interbasin 9259 has multiple upstream areas. Focusing on area 9259, the upstream areas have codes equal to 9259+2 and 9259+12

Bibliography

- Barrow, C., 1998. River basin development planning and management: A critical review. World Development 26 (I), 171-186.

- Burton, J., 1995. A framework for integrated river basin management. Water Sciences and Technology 32 (5-6), 139-144.

- EROS-1, 2001. HYDRO1K website:
  http://edcdaac.usgs.gov/gtopo30/hydro/
  Available as of 11/26/01

- ESRI, 2000. ArcDoc Version 8.1 Pre-release. ArcINFO Help

- ESRI, 2001 GIS User Conference, Pre-Conference Seminar on ArcGIS Hydro Data Model

- ESRI-1, 2001. ArcGIS Version 8.1, ArcGIS Help Manual

- Franken, S., Tyler, D., Verdin, K. 2001. "Development of a National seamless database of topography and hydrologic derivatives." 2001 ESRI GIS User Conference Preceedings.

- Furnans, J., Olivera, F. 2001. "Drainage Systems." Chapter 4 of ArcGIS Hydro Data Model Documentation. ESRI, Inc. Press.

- Furnans, J., Olivera, F. 2002. "Hydrologic Data Development for South East Asia and Latin America." Report to be submitted to the Food and Agricultural Organization of the United Nations. Will be available at http://www.crwr.utexas.edu/online.shtml

- Goulter, I., Forrest, D., 1987. Use of geographical information systems (GIS) in river basin management. Water Science and Technology 19 (9), 81-86.

- Jensen, S., Domingue, J., 1988. Extracting topographic structure from digital elevation data for geographic information system analysis. Photogrammetric Engineering and Remote Sensing 54, 1593-1600.

- Maidment, David R. 2001. ArcGIS Hydro Data Model ESRI. Inc. Press.

- Newson, M., 1992. Land, Water, and Development: River Basin Systems and their Sustainable Management. Routledge, London. 351 pp.

- Pfafstetter, Otto., 1989. Classification of hydrographic basins: coding methodology, unpublished manuscript, Departamento Nacional de Obras de Saneamento, August 18, 1989, Rio de Janeiro; available from J.P. Verdin, U.S. Geological Survey, EROS Data Center, Sioux Falls, South Dakota 57198 USA.

- Roche, M., 1968. Traitment automatique de donnees hydrometriques et des donnes pluviomtriques au service hydrologique de l'ORSTOM, Cahiers de l'ORSTOM, Serie Hydrologique. Vol. V, no. 3; ORSTOM, 209 rue La Fayette, 75480 Paris, France.

- Seaber, P.R., Kapinos, F.P., and Knapp, G.L., 1987, Hydrologic Unit Maps: U.S. Geological Survey Water-Supply Paper 2294, 63 p.

- Silva, Paulo Afonso. 1999. "Classificação e Codificação das Bacias Hidrográficas Brasileiras Segundo o Método Pfafstetter, Com Uso de Geoprocessamento." Proceedings from *II-Encontro Das Aguas –* Montevideo, Uruguay. Available at: http://www.iica.org.uy/16-6-pan1-pon10.htm as of 11/26/01

- USDA: National Watershed Boundary Dataset Website: http://www.ftw.nrcs.usda.gov/huc_data.html - Available as of 11/26/01

- USGS-1, 2001. Hydrologic Units website: http://water.usgs.gov/GIS/huc.html Available as of 11/26/01

- USGS-2, 2001. DEM Generation website : http://edcwww.cr.usgs.gov/nsdi/gendem.htm Available as of 1/31/01

- USGS-3, 2001. NWIS Website: http://water.usgs.gov/nwis/ available 11/26/01

- Verdin, Kristine L. "A System for Topologically Coding Global Drainage Basins and Stream Networks". 1997 ESRI International GIS User Conference Proceedings

- Verdin, K.L. and J.P. Verdin, 1999, A topological system for delineation and codification of the Earth's river basins," *Journal of Hydrology*, vol. 218, nos. 1-2, pp. 1-12

- Verdin, K., 2001. Personal communication via email, 8/26/00-11/26/01

- Wahl, K.L., Thomas, W.O., Jr., Hirsch, R.M., 1995. Stream-Gaging Program of the U.S. Geological Survey, U.S. Geological Survey Circular 1123, Reston , Virginia.

VITA

Jordan Ernest Furnans was born in Massachusetts on the 24th of June, 1977, the son of Mark Jay Furnans and Karen Ruddeforth Furnans. After completing his work at William H. Hall High School, West Hartford, CT, in 1995 he entered Princeton University in Princeton, NJ. During the Spring of 1997 he attended the University of St. Andrews in St. Andrews, Scotland. He received the degree of Batchelor of Science in Engineering from Princeton University on June 1, 1999. He also received certificates of merit in Geological Engineering and Spanish. During the following summer, he worked for Princeton University Outdoor Action and managed wilderness excursions for over 600 Princeton University students. In August, 1999, he entered The Graduate School at The University of Texas at Austin. Mr. Furnans will spend the twelve-month period between June 2002 and June 2003 studying with Dr. Jorg Imberger at the Centre for Water Research at The University of Western Australia under a prestigious Fulbright fellowship.

Permanent Address: 131 Mohegan Dr.
West Hartford, CT 06117

This thesis was typed by the author.